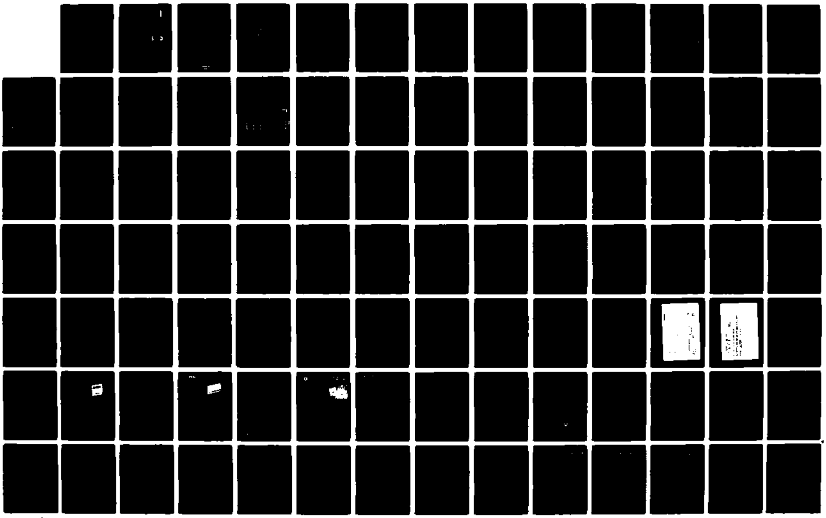
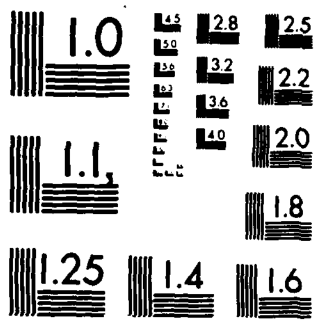
AD-A128 338 MULTIPLE-FREQUENCY ULTRASONIC PULSE-ECHO DISPLAY SYSTEM 1/2
(U) DESIGN AUTOMATION INC LEXINGTON MASS 28 SEP 82
DAAG46-80-C-0010

UNCLASSIFIED F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

PHOTOGRAPH THIS SHEET

A128 338

DTIC ACCESSION NUMBER

II

LEVEL

INVENTORY

Multiple-Frequency Ultrasonic Pulse-Echo Display System. Final Tech Rept.

26 Jun. '81

Revised 28 Sept. '82

Contract DAAG46-80-C-0010

DOCUMENT IDENTIFICATION

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

ACCESSION FOR		
NTIS	GRA&I	☒
DTIC	TAB	☐
UNANNOUNCED		☐
JUSTIFICATION		
BY Per Ltr. on File		
DISTRIBUTION / FL-86, Acq. #83-0563		
AVAILABILITY CODES chd. 19 Apr '83		
DIST	AVAIL AND/OR SPECIAL	
A	23 / 2	

DISTRIBUTION STAMP

DTIC
ELECTE
MAY 16 1983
S D
D

DATE ACCESSIONED

DTIC COPY INSPECTED

"Original contains color plates: All DTIC reproductions will be in black and white"

83 05 13 072

DATE RECEIVED IN DTIC

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-DDA-2

DTIC FORM OCT 79 70A

DOCUMENT PROCESSING SHEET

REVISED

DESIGN AUTOMATION, INC.

809 MASSACHUSETTS AVENUE • LEXINGTON, MASS. 02173
TELEPHONE (617) 862-8998

AD A128338

MULTIPLE-FREQUENCY ULTRASONIC PULSE-ECHO DISPLAY SYSTEM

FINAL TECHNICAL REPORT

CONTRACT ITEM A002

Contract No.: DAAG46-80-C-0010
Army Materials and Mechanics Research Center
Watertown, Massachusetts 02172

Project 4285

June 26, 1981
Revised September 28, 1982

DISCLAIMER NOTICE

MULTIPLE-FREQUENCY ULTRASONIC PULSE-ECHO DISPLAY SYSTEM

TABLE OF CONTENTS

I. PROJECT SUMMARY

The object of the Multiple Frequency Ultrasonic Pulse-Echo Display System is to provide an enhanced graphic display of the response of materials under test, to ultrasonic pulse excitation. Echo responses in three frequency bands are displayed in three respective colors. The width and center frequency of each band are individualy adjustable, with an overall frequency span from 1 to 10 MHz. The system also provides the capability for scaling waveforms and for permanent data storage.

All of the original objectives of the development programs have been met. Off-the-shelf devices provide the functions of transducer, pulser, amplifier, display, and storage. Specialized hardware has been built to filter and detect the signal waveforms, provide a true stepless gate, and encode all relevant data in a suitable form to be displayed. Custom software was written for a commercial terminal to provide three-color display and storage capabilities for measurement data, and to allow the displayed waveforms to be scaled to any desired level. Moreover, the waveforms can be represented in either linear or logarithmic form.

The color-graphics computer terminal which provides storage and display capabilities, is limited somewhat in speed; display of a single set of waveforms (called a "snapshot") can take several minutes. While this is not objectionable for close scrutiny of a set of waveforms or for storage, it is inconvenient for setting up the desired display. Therefore, a real-time oscilloscope display is included in the system, along with certain custom hardware which facilitates its operation. The oscilloscope displays the detected

output of all three frequency bands. An intensified trace identifies the time window selected by the stepless gate, and a limiter provides non-ambiguous indication of excessive input levels.

The final system permits rapid setup, and accurate measurement of both amplitude and time.

I SYSTEM DESCRIPTION

A. Overall Description and Interconnection

As shown in Fig. II-1, the system is divided into four major functional components: Pulser/Receiver, Filter/Digitizer, oscilloscope display and graphics computer.

The Pulser/Receiver block includes a wideband ultrasonic transducer. Both the transducer and the pulser/receiver are standard products of Panametrics, Inc., Waltham, MA. In response to 1 kHz TRIGGER pulses from the Filter/Digitizer, the Pulser/Receiver generates a high-energy short-duration impulse which drives the transducer. The impulse ("main bang") and the response from the transducer and test sample are amplified and sent to the Filter/ Digitizer via the SIGNAL OUT connector.

In the Filter/Digitizer the RF signal is applied to the inputs of three variable filters via independent adjustable attenuators. The output of each filter is peak-detected; the detected video signals are made available at the DEMOD OUT connectors. These outputs drive three of the four oscilloscope vertical channels, for a real-time display of the filtered and detected pulse-echo responses. In addition, the Filter/Digitizer provides the function of a stepless gate. Both the delay and duration of the gate are adjustable. A signal corresponding to the gate pulse is available at the DELAY connector. This signal drives the fourth oscilloscope vertical channel and the Z-axis input as well, so that the waveforms are intensified during the gated time window. The oscilloscope can be triggered on the start of the gated time window or the "main bang."

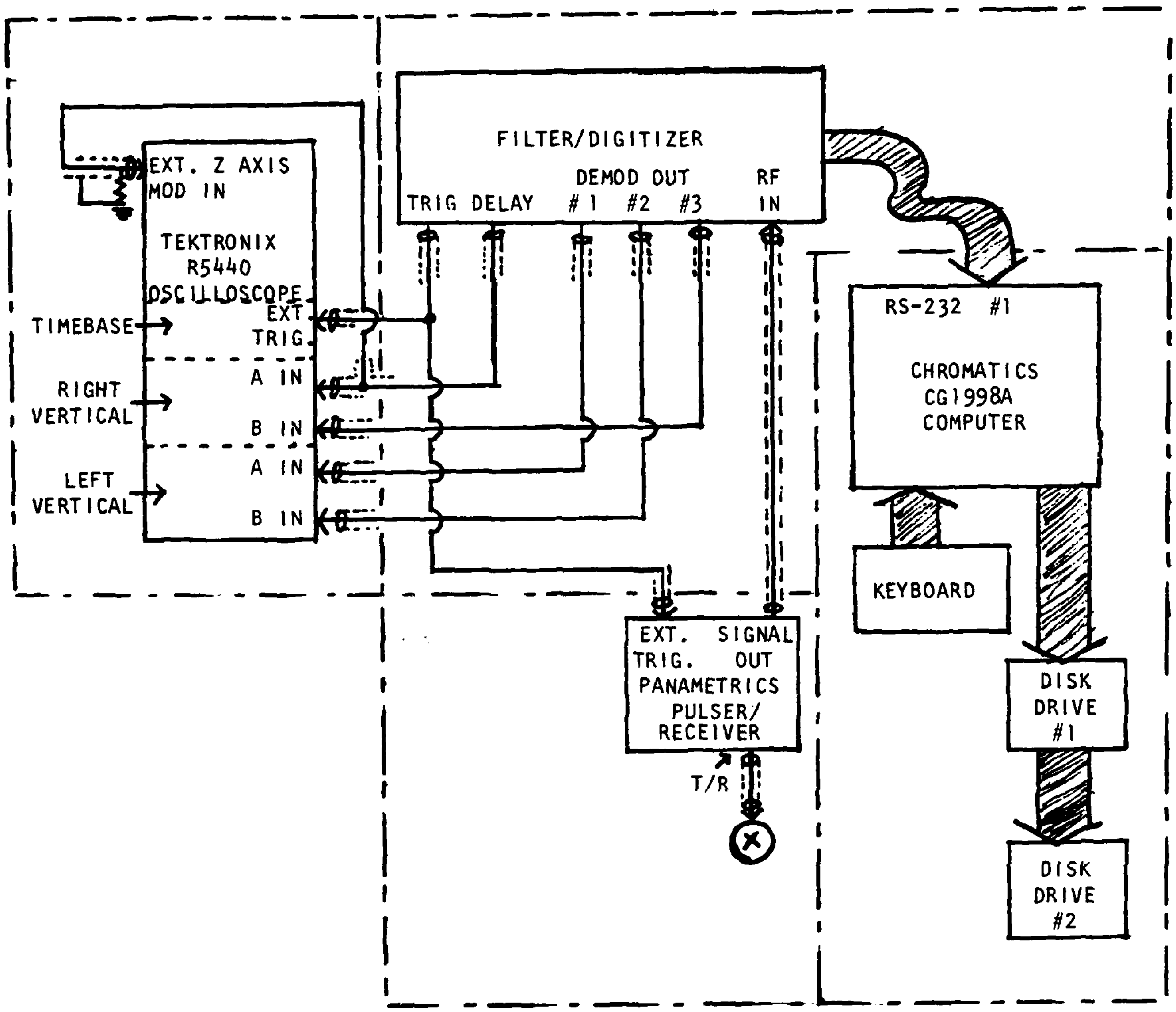


Fig. II-1. System interconnection diagram.

The gated waveforms, as well as the settings of the various Filter/Digitizer controls (FREQUENCY, BANDWIDTH, LIN/LOG, DELAY TIMEBASE) are encoded into digital form and transmitted to the Chromatics color graphics computer. Associated with the computer are a keyboard and two disk drives. The computer displays and processes individual "snapshots" of data, each consisting of 448 samples of the waveform from each filter channel. "Snapshots" may be acquired, scaled, stored, and retrieved und computer control once the data is stored in the terminal memory

B. Pulser/Receiver

The pulser/Receiver is a Panametrics type 5055PR; the transducer is a Panametrics type V109. Detailed specifications of these devices can be found in the separate Panametrics manual.

C. Oscilloscope

The oscilloscope consists of a Tektronix 5440 rackmount oscilloscope mainframe, two 5A48 dual trace vertical amplifiers, and a 5B42 delaying timebase. This configuration provides 50 MHz bandwidth and triggering over the full bandwidth. Full specifications can be found in the manuals for the oscilloscope. When connected into the system (see Fig.II-1), this scope gives a real-time display of the filtered and detected pulse-echo response. The brightened trace shows clearly the waveforms which would be displayed on the computer if a "snapshot" were taken at a given instant. The displayed waveforms limit sharply at the maximum permissible signal level. One should remember that the oscilloscope controls, unlike

those of the Filter/Digitizer, have no effect on the waveforms displayed by the computer. The oscilloscope can be thought of as a "viewfinder" to the "camera" formed by the Filter/Digitizer and computer, to be used in "framing" and "focusing" the "snapshots."

D. Filter/Digitizer

The Filter/Digitizer provides three primary functions: timing and gating, analog signal processing, and digital data conversion and transmission. A simplified block diagram of the unit is shown in Fig.II-2.

A TRIGGER pulse is sent to the Pulser/Receiver every millisecond. The pulse-echo response (SIGNAL OUT from the Pulser/Receiver) is applied to the inputs of the three variable filters. The signal level at each input is individually adjustable via a four-turn potentiometer, with turns-counting dial. The filter outputs are rectified and smoothed by a full-wave detector. The outputs of the detectors, after being amplified and buffered, are available at the rear panel, to be monitored by the oscilloscope. The detector outputs form the inputs to the multiplexer, The output of the latter is connected in turn, to the sample-and-hold amplifier. Both the sample-and-hold amplifier (S/H) and multiplexer are controlled by the timing and control generators. For each channel in turn, 448 samples are taken, one every millisecond. The multiplexer is then switched, and the next channel is sampled 448 times. Sampling and conversion of the output of all three channels therefore takes 3 x 448 - 1344 ms. Accurate control of the sample timing provides the equivalent function of a stepless gate. The output of the sample-and-hold unit is applied

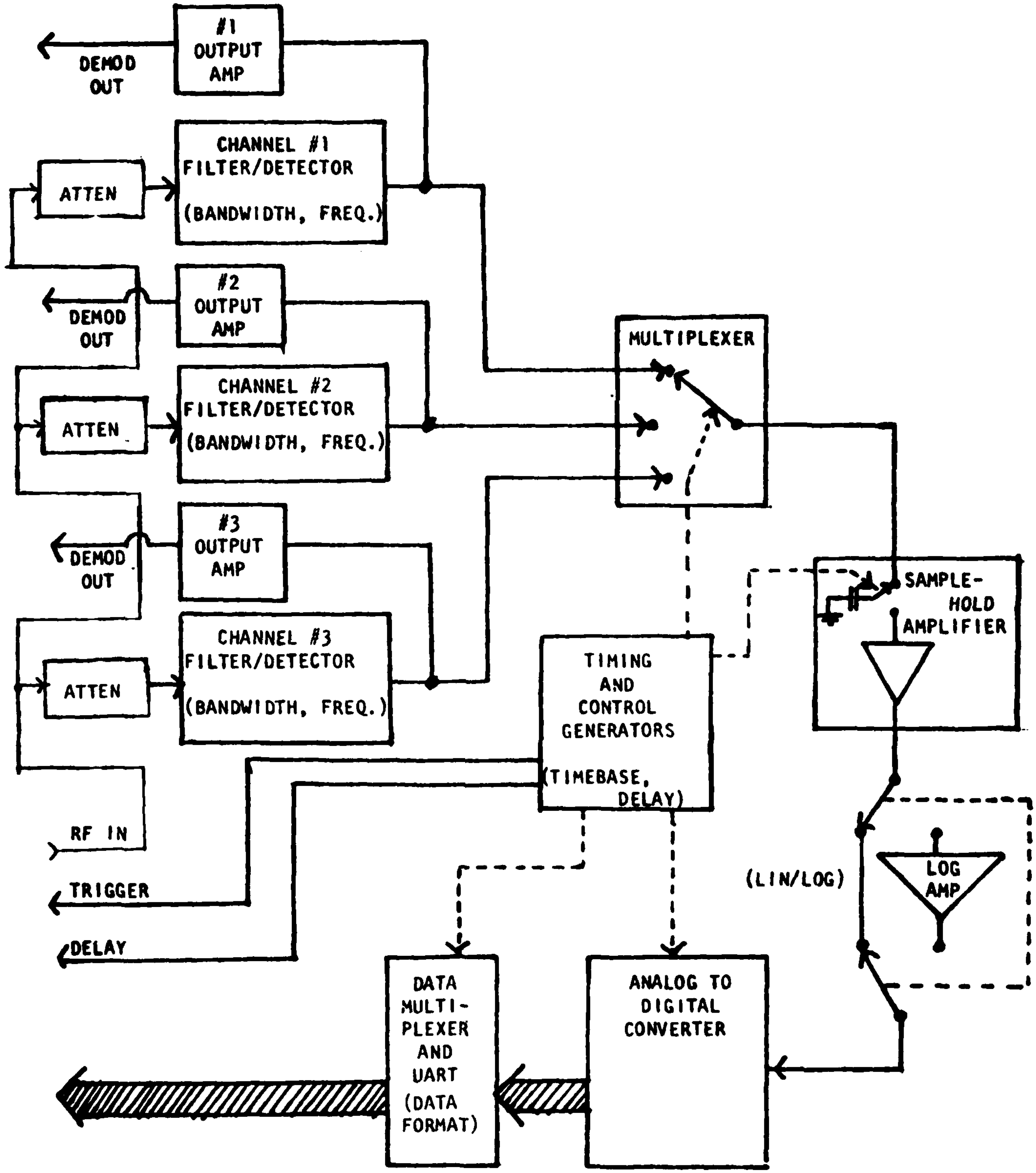


Fig.II-2. Block diagram of Filter/Digitizer. Solid lines are signal paths; dotted lines are timing and control connections. Parenthesis indicate front panel controls.

to the analog-to-digital converter (ADC), either directly, or through a logarithmic amplifier. The ADC output, as well as the encoded positions of all switches and controls on the Filter/Digitizer are sent to the UART (Universal Asynchronous Receiver Transmitter), a standard integrated circuit which is used as the interface between the Filter/Digitizer and the computer. The format of the output data from the UART can be switched between two options, as detailed in Section III-H.

III. FILTER/DIGITIZER CIRCUIT DESCRIPTION

A. Overall

A simple block diagram of the Filter/Digitizer is shown in Fig. 2. The functional block diagram (Fig.B-1) permits the operation of this device to be analyzed in some detail.

The circuitry is functionally divided into ten circuit boards, which are mounted in a card file and interconnected via backplane wiring. The power supply is mounted on the back of the card file. All power supply connections are made through board #10. Connections to the front panel are through board #6. Boards #1, 2, and 3 contain the filters and detectors for processing of the pulse-echo signal. Board #4 is a digital data acquisition system, which converts the outputs of boards #1, 2, and 3 to digital form in either linear or logarithmic representation. Timing for the conversion process is controlled by boards #5, 7, and 8. On board #9, the digitally encoded waveforms and the settings of all the Filter/Digitizer front panel controls are multiplexed, and transmitted to the computer in serial form.

B. Boards #1, 2, and 3 (Filter/Detectors)

Each of these boards is configured as shown in Fig. III-1. The filter input impedance is high (>500Ω) so that the rf connection is a short 50Ω unterminated coaxial cable connection from the wiper of a precision potentiometer. The three potentiometers are connected in

parallel across RF IN. The combined impedance of the potentiometers and filter input stages provides the 50Ω termination for RF IN; and the four-turn potentiometers permit accurate, independent attenuation of the inputs for each of the three channels. The boards are therefore electrically interchangeable, although switching board positions will cause front panel and computer displays of filter frequency to be in error. On each board there is a shorting plug connecting the buffer output and the full-wave detector input. This connection can be used for troubleshooting. The three boards are interconnected as shown in Fig. III-2.

The bandpass filter circuit used in each channel is shown in Fig. III-3. For narrow bandwidths, high quality reactive components are required. L2 is therefore an air-core inductor and C_0 through C_7 are silver-mica capacitors. L1 is a ferrite-core RF choke, and presents a very high impedance at the frequencies of interest. C10 is present for DC blocking only. The DC operating point of Q1 is set by R1-R4. R5 and R6 are coupled so that the gain of this stage at the center frequency is essentially constant over the range of the bandwidth adjustment. R7 is a gain trimming potentiometer, which is set to minimize the difference in gain between bandwidth control settings. The output at Q1's collector is buffered by Q2 which drives the input of the detector stage. The positions of the FREQUENCY and BANDWIDTH control settings are detected and encoded by Z1 and Z2 (see overall schematic). The outputs of these encoders are connected to board #9, to be transmitted to the display terminal.

The circuit of the detector stages is shown in Fig. III-4. It consists of two differential amplifiers, A1 and A2, a push-pull current amplifier, and a bridge rectifier. Using a current amplifier to drive the rectifier provides a linear high-frequency

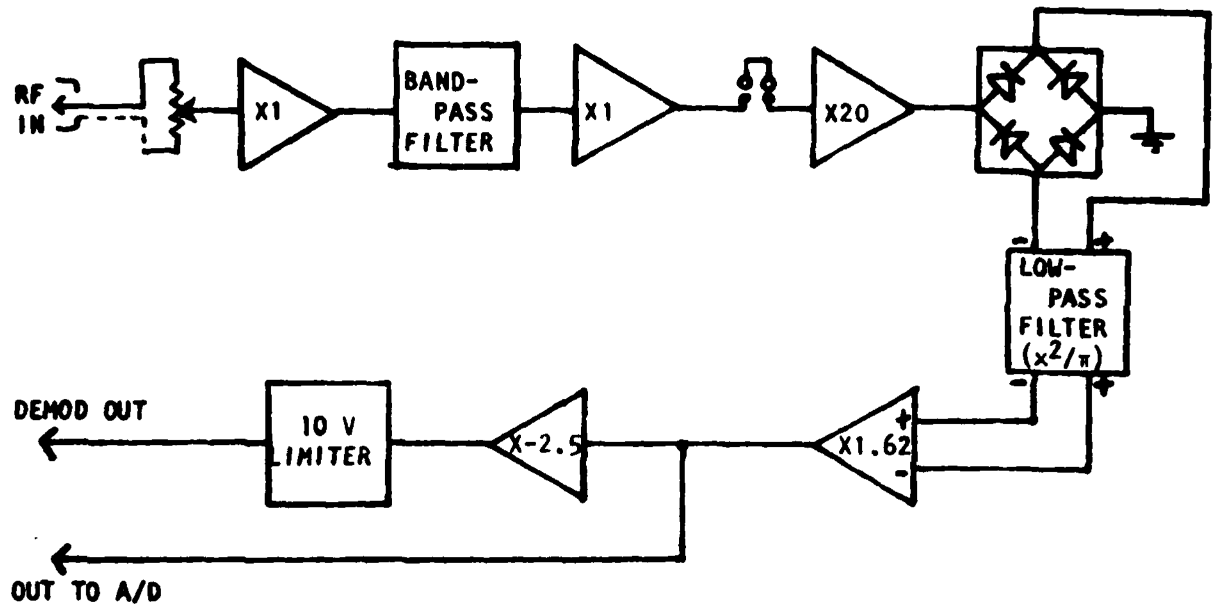


Fig. III-1. Block diagram of boards #1, 2, and 3.

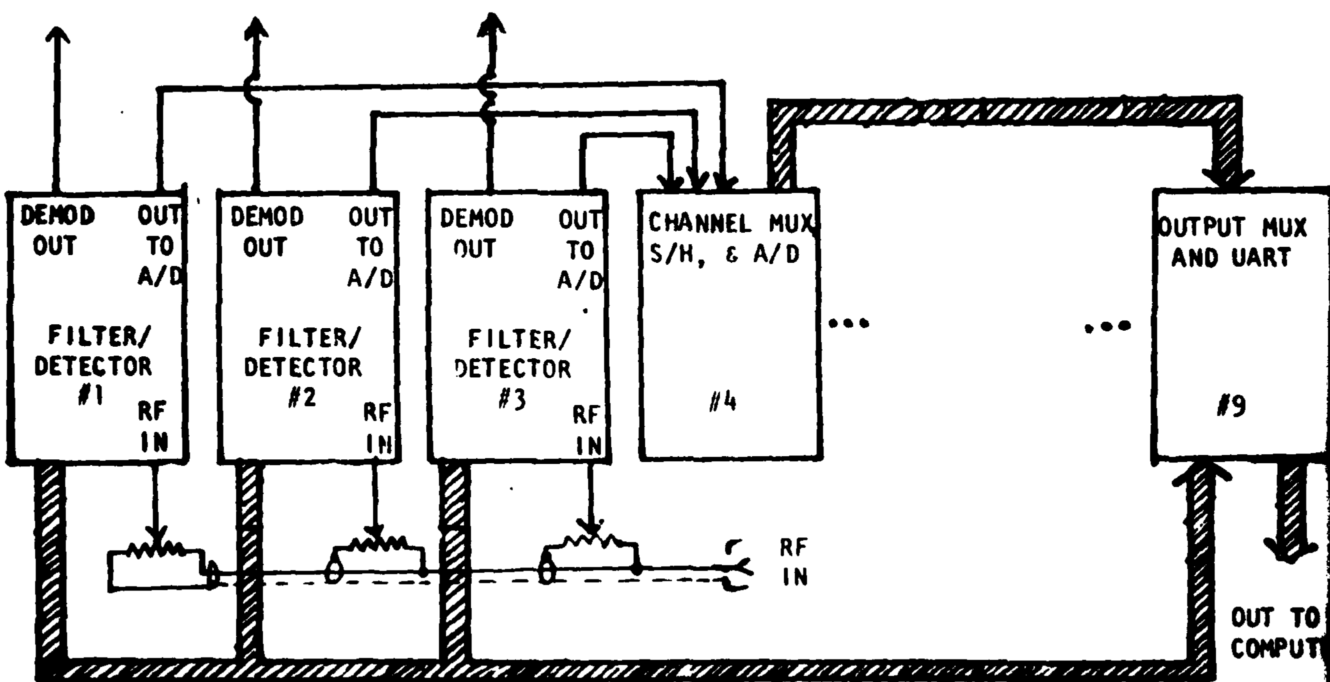


Fig. III-2. Signal and data flow chart.

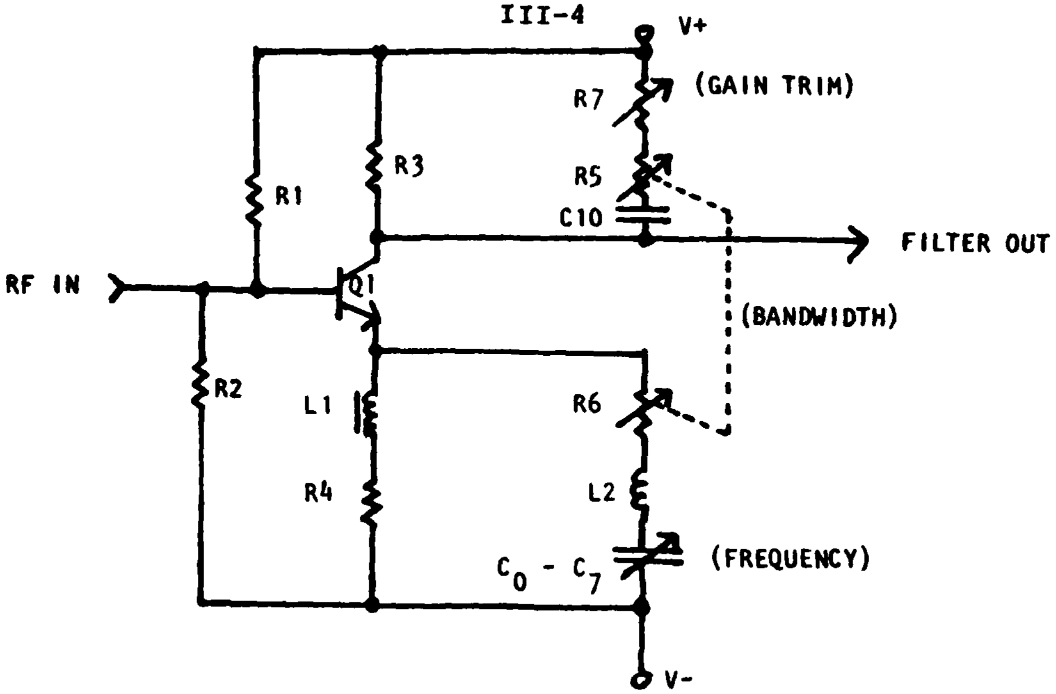


Fig. III-3. Bandpass filter circuit.

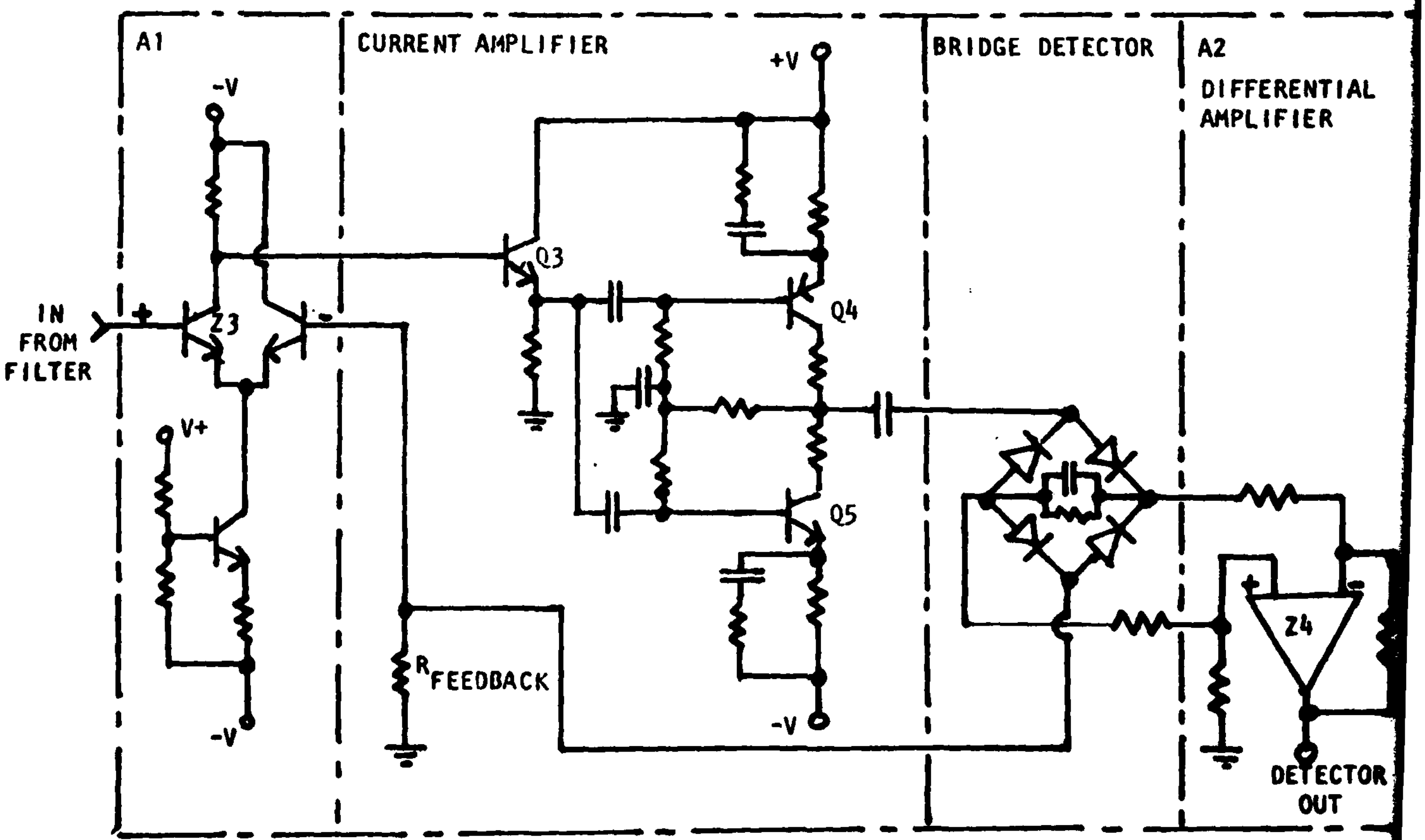


Fig. III-4. Linear full-wave detector circuit topology.

response and large dynamic range. Feedback around the A1-current amplifier-rectifier loop further enhances these characteristics. The full-wave rectified video signal is converted from differential to single-ended form by op-amp Z4. Z4's output (a unipolar negative signal) is connected to the input of the inverter/limiter, and to the input of the channel multiplexer (board #4). The limit level adjustment (see overall schematic) should be set to limit the output of the inverter/limiter at + 10V, This will make the limit levels of both oscilloscope and computer displays identical.

BOARD #1, #2, #3 ADJUSTMENTS SUMMARY

NAME	EQUIPMENT NECESSARY	PROCEDURE
C_{trim}	Sweep generator to cover the range of 0.5-12 MHz. 50 MHz oscilloscope	Set the sweep generator output level to 0.2 V p-p. Set up oscilloscope and sweep generator for swept frequency measurements. Adjust horizontal display to give: 0.25 MHz/div. for board #1, 0.5 MHz/div for board #2, and 1 MHz/div for board #3. Connect oscilloscope verti-cal drive to the channel output on the front panel. Set bandwidth adjustment to 0.25 MHz. Adjust C_{trim} to give 1 divisio difference between center frequencies of adjacent frequency select positions, espe-cially at the highest frequencies. Set all attenuators to 0.00.
gain trim	Same as for C_{trim}	Same connections as above. Set frequency control on the Filter/Digitizer channel of interest to midrange. Note change in gain at the center frequency as the band-width control is switched over its range. Adjust gain trim potentiometer to minimi this gain variation.
limit level	Same as for C_{trim}	Increase output level of sweep generator to approximately 0.5 V p-p. Set limit level potentiometer so output clips at + 10.0 V.

C. Board #4 (Channel Multiplexer, S/H, and ADC)

This board is a fairly conventional digital-data acquisition system, with a few special features. The block diagram is shown in Fig. III-5. A 12-bit analog-to-digital converter (ADC) is used for high accuracy and wide dynamic range, although only 11 bits are actually transmitted to the computer (nine bits with the revised data format). By preceding the ADC with a sample-and-hold (S/H) amplifier, and requiring only one conversion every millisecond, we are able to use a relatively slow ADC. The S/H amplifier, however, must be very fast and jitter-free. Since the shortest timebase is 1 μs, and 448 samples are taken to represent this interval, the timing accuracy of the sample-and-hold circuitry must be better than 1 μs/448 = 2.3 ns (even though the samples are taken 1 ms apart). The relevant specification for the chosen S/H amplifier-aperture uncertainty time - is 250 ps, providing a factor of 10 margin.

Fig. III-6 is the timing diagram corresponding to Fig. 5. Since, as seen in the timing diagram, the ADC begins its convert cycle 900 μs after the pulser is triggered, return signals occurring after this time cannot be acquired. The channel-select counter is updated every 448 ms. The output of the ADC is sent to board #9, where the 11 most significant bits are transmitted to the computer. The state of the channel-select counter is also transmitted, to identify the data. Note that valid data to be transmitted during a given 1 kHz clock period corresponds to a sample taken during the previous clock period.

One way in which Board #4 differs from most digital data acquisition systems is in the presence of a switchable logarithmic amplifier. By computing the logarithmic of a signal and then converting the

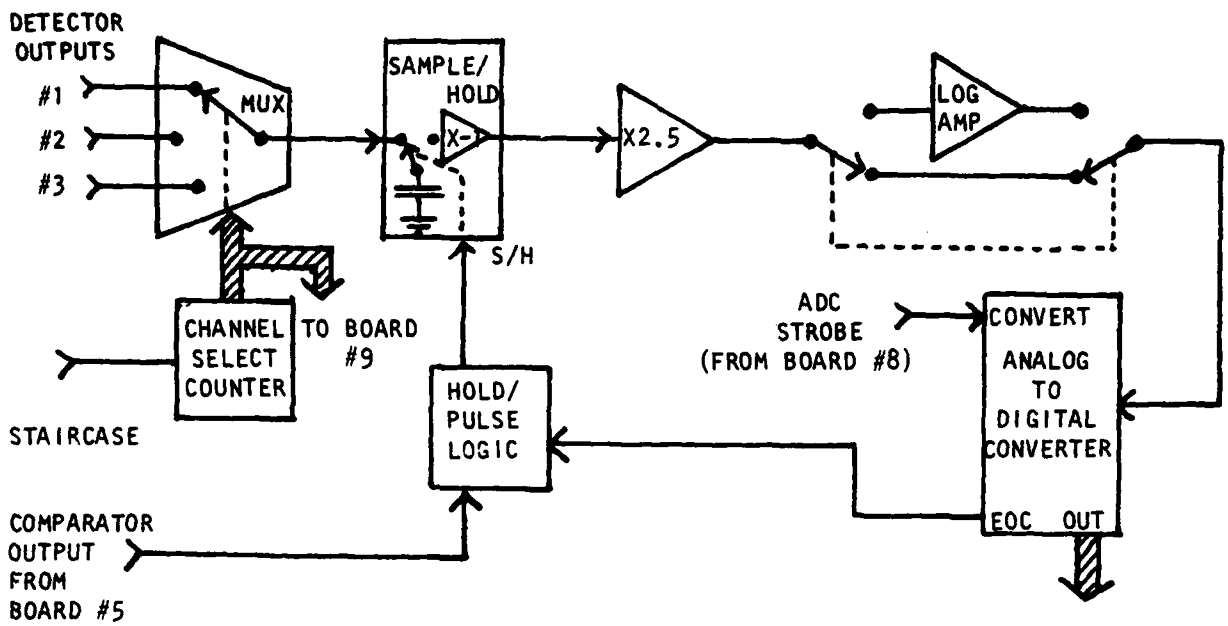


Fig. III-5. Block diagram of Board #4.

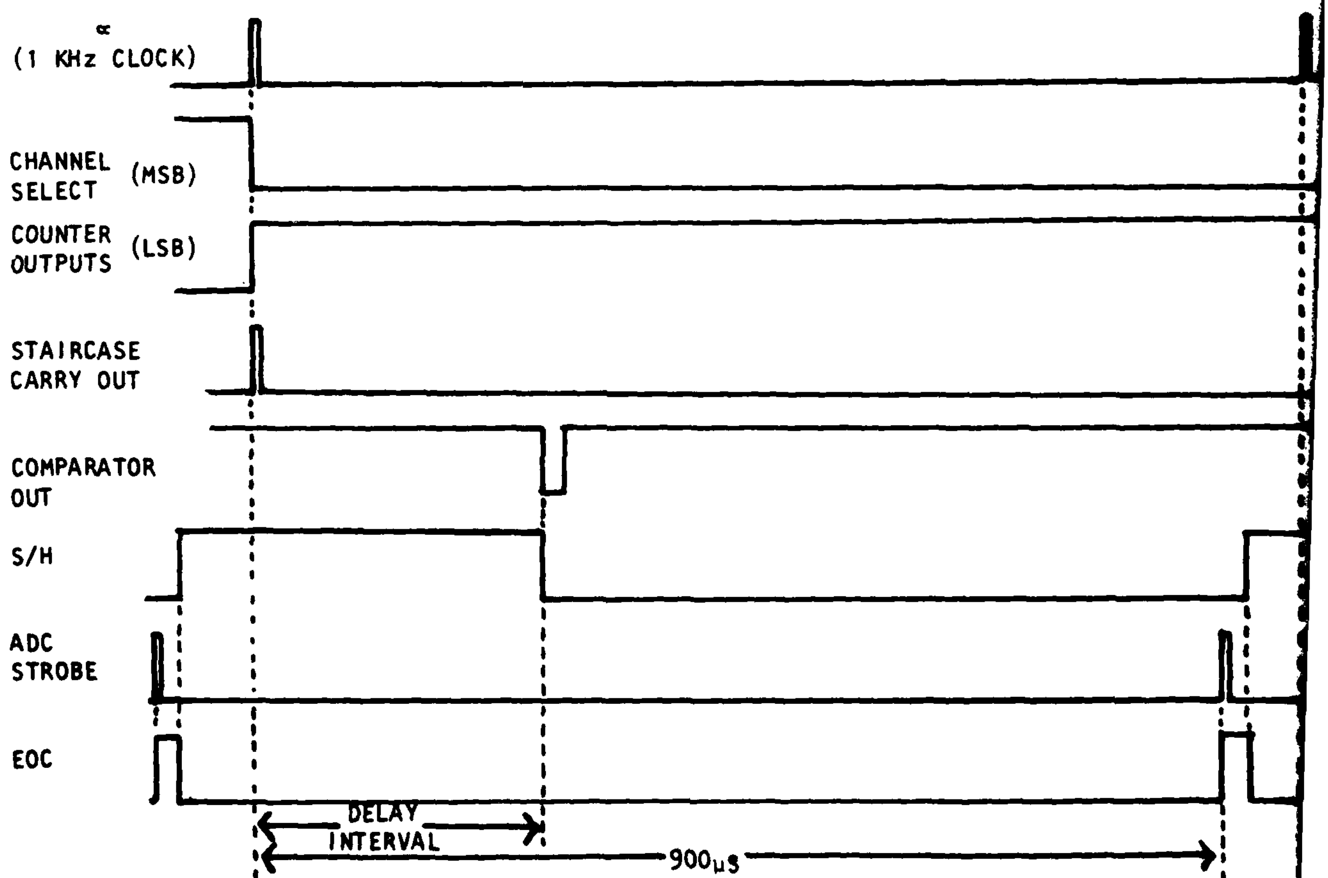


Fig. III-6. Board #4 timing.

logarithm to digital form, one would obtain a much greater dynamic range than by direct digital conversion of the signal. While accurate logarithmic amplifiers are not available with the necessary operating speed, by first sampling the signal and then computing its logarithm we are able to realize an equivalent result. The logarithmic unit covers four decades, although system noise (primarily from the Pulser/Receiver) limits usable dynamic range to no more than three decades.

Board #4 ADJUSTMENT SUMMARY

NAME	EQUIPMENT NECESSARY	PROCEDURE
Log Amp Offset	3½ digit digital voltmeter	Remove Z8. Connect a jumper between pin 4 of the log amp (Z9) and ground. Monitor voltage at pin 3 of log amp. Adjust potentiometer until output voltage is between +8 and +10 V.
Log Output Offset	3½ digit DVM, 0-10 V adjustable voltage source	Set voltage source to +0.1V. Connect to input (pin 4) of log amp. Output (pin 3) should be near 0V. Adjust voltage source to give exactly 0V at pin 3. Adjust log output offset potentiometer for +4 V at pin 6 of Z10. Replace Z8.
Sample and Hold Offset	Same	Remove Z1. Connect a jumper between pins 31 and 32 of the sample/hold amplifier (Z2). Set Lin/log switch to log. Monitor the A/D outputs (pins 5-16 of Z3). Adjust the S/H offset potentiometer to give all outputs= "0" except the least significant bit (pin 1a of Z3), which should flicker between "1" and "0". Replace Z1.
ADC Gain Adjust	Same as above	Remove Z8. Set voltage source to 10.00 V and connect to input of A/D (pin 32 of Z3). Monitor outputs (pins 5-16 of Z3). Set gain adjust potentiometer so all outputs are "1", except the least significant bit (pin 16 of Z3), which should flicker between "0" and "1". Replace Z8.

D. Board #5 (Fast-Ramp Generator)

Board #5 generates a negative-going fast ramp every millisecond, at the end of the variable delay interval, in response to the $\overline{\text{TRIG}}$ pulse. By varying C_T, the duration of this ramp is switch-selectable in six steps between 1 μs and 50 μs. Z2 (see Fig. B-4) encodes the TIMEBASE switch position and transmits the data to the output multiplexer and UART (board #9). The fast ramp is compared to a 448-step negative going staircase signal by the high-speed comparator (Z3). The output of this comparator is a negative transition every millisecond. These transitions are the timing reference for the S/H amplifier (on board #4) and thus provide 448 samples equally spaced in time over the interval defined by the fast ramp. Fig.III-7 is a rough schematic, illustrating circuit operation. The components peripheral to the comparator scale the fast ramp and staircase waveforms to appropriate levels, and establish a small amount of hysteresis to aid in stabilizing the comparator. Fig. III-8 illustrates the timing of the waveforms involved in the sampling process.

Auxiliary logic (Z4) and an additional comparator (Z6) are used to generate the DELAY pulse. The delay pulse is high during the interval when the signal is sampled. Z5 is used to equalize delays between the sampling interval and the DELAY pulse output, so that the oscilloscope display corresponds exactly to the acquired data.

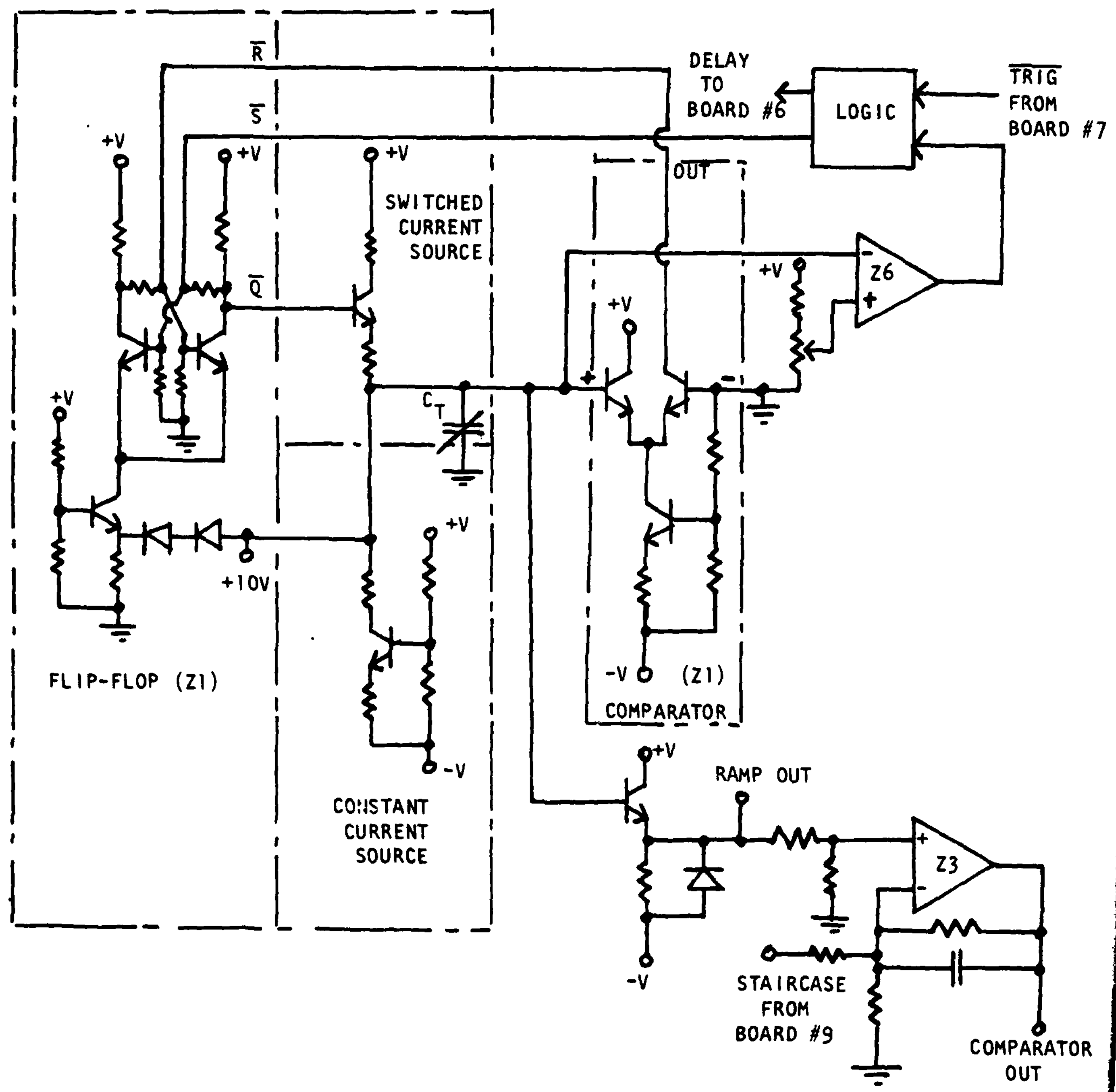
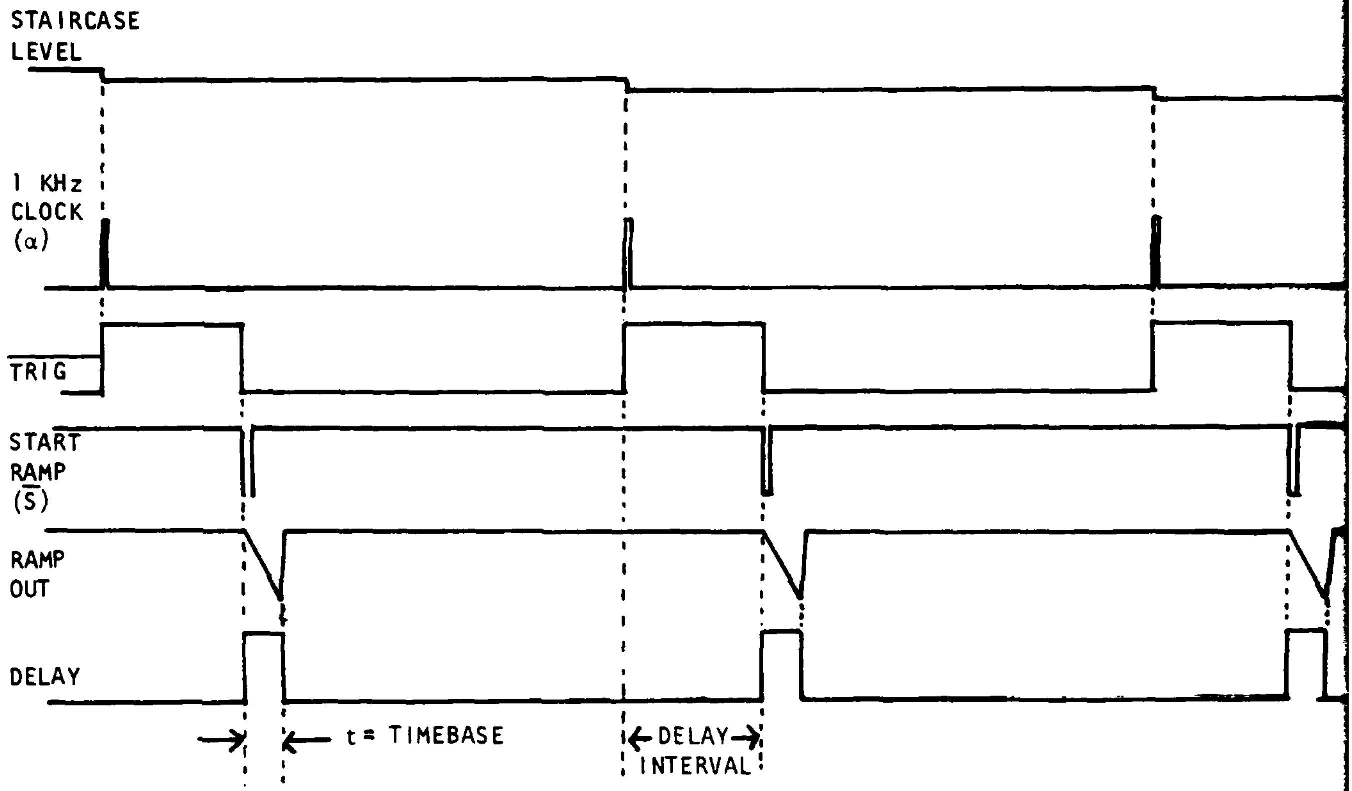
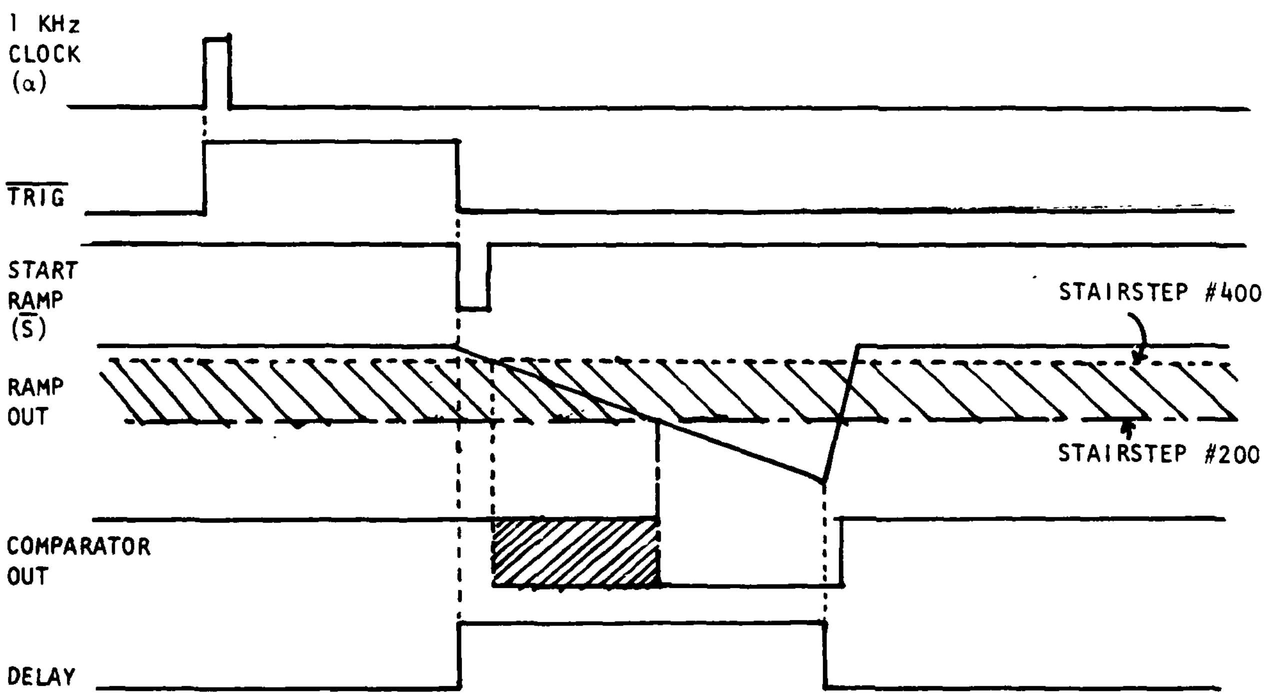


Fig. III-7. Functional schematic of board #5.



(a)



(b)

Fig. III-8. Sampling and DELAY timing and waveforms.

Board #5 ADJUSTMENTS SUMMARY

NAME	EQUIPMENT REQUIRED	PROCEDURE
Ramp Rate	50 MHz oscilloscope	Trigger scope from $\overline{\text{TRIG}}$ (pin 12 of Z4). Monitor the sample pulse output (pin 4 of Z3). The negative transition is of interest. Set TIMEBASE control to 10 μs. Every 0.5 s the sample pulse negative transition will sweep across some time interval. Adjust the ramp rate potentiometer to set this interval to exactly 10 μs.
Ramp Delay	None	Set time base to 1.0 μs. Adjust DELAY controls for >10 μs of delay. Take a snapshot. Check that the start of the intensified sweep on the oscilloscope corresponds closely to the start of the displayed snapshot. Adjust the ramp delay potentiometer to give a correspondence better than 0.1 μs.
Delay Duration	None	Set timebase to 50 μs. Adjust delay duration potentiometer to set intensified sweep length to 50 μs ±1 μs.

E. Board #6 (Panel Interface)

Board #6 is hard-wired to the front panel. They can be removed as a unit by removing the front panel, unplugging the two Cinch-Jones® line connectors, disconnecting the rf input connections to boards #1, #2, and #3, and unplugging board #6. To remove the front panel: remove all the front panel knobs (using a 1/16" allen wrench) except the lin/log and format control knobs, remove each of the two screws from either end of the front panel, and slide the panel straight out. Do not pull board #6 by the wires to remove it. The front panel fuse is a 2A slo-blo type.

F. Board #7 (Delay Generator)

In the Filter/Digitizer, the operator selects the desired delay interval by depressing any of the three increment/decrement "DELAY" switches on the panel while observing the echo waveform on the oscilloscope in relation to the intensified gate interval. The "Delay" switches, when activated, cause the delay-set counter on Board #7 to increment or decrement until a desired value is achieved. The three switches are labelled "Fast", "Slow," "Single-Step," and represent the rate at which the delay-set counter increments or decrements when they are activated. (Moving a switch up increments the counter; moving it down decrements the counter.) The number ultimately stored in the delay-set cou[illegible] represents the delay between the "man bang" and the start of the gate interval of interest. This interval is represented by the pulse $\overline{\text{TRIG}}$.

The principal purpose of Board #7 is to derive the pulse ($\overline{\text{TRIG}}$), whose rising edge coincides with the pulser trigger, and whose width spans the variable accurately-controlled delay interval: The delay is set by clocking the input of a four-digit BCD counter, till its count (expressed as 100's of nanoseconds) reaches the desired value. (See Fig.III-9). The maximum delay is therefore 999.9 μs.

The rate at which this delay-set counter is clocked is determined by the choice of DELAY switch. In the "Fast" position, the counter uses a 1 KHz clock giving 100 μs per second change in delay. In the "Slow" position, the counter responds to a 30 Hz clock (3.3 μs/s),

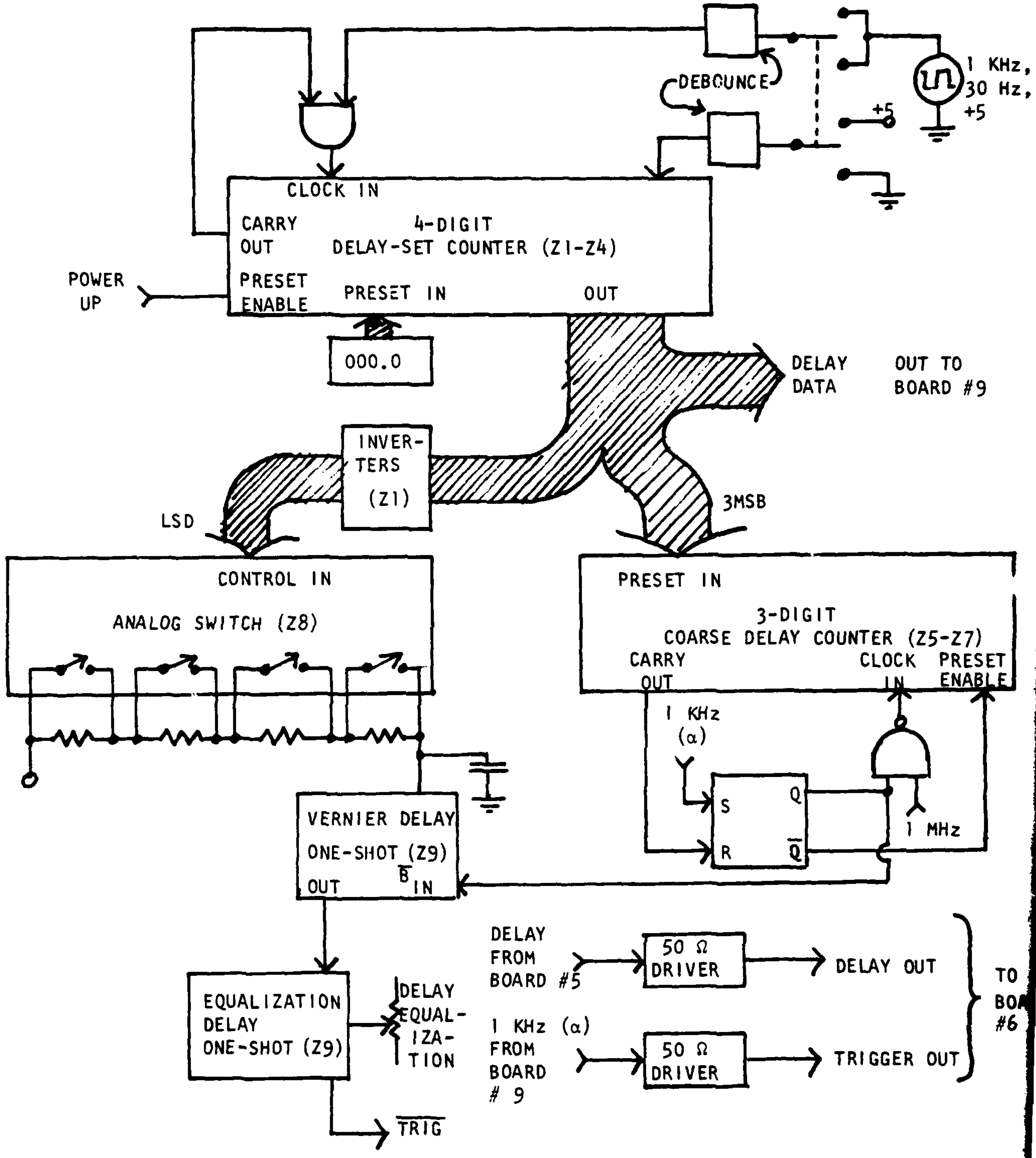


Fig. III-9. Functional diagram of board #7.

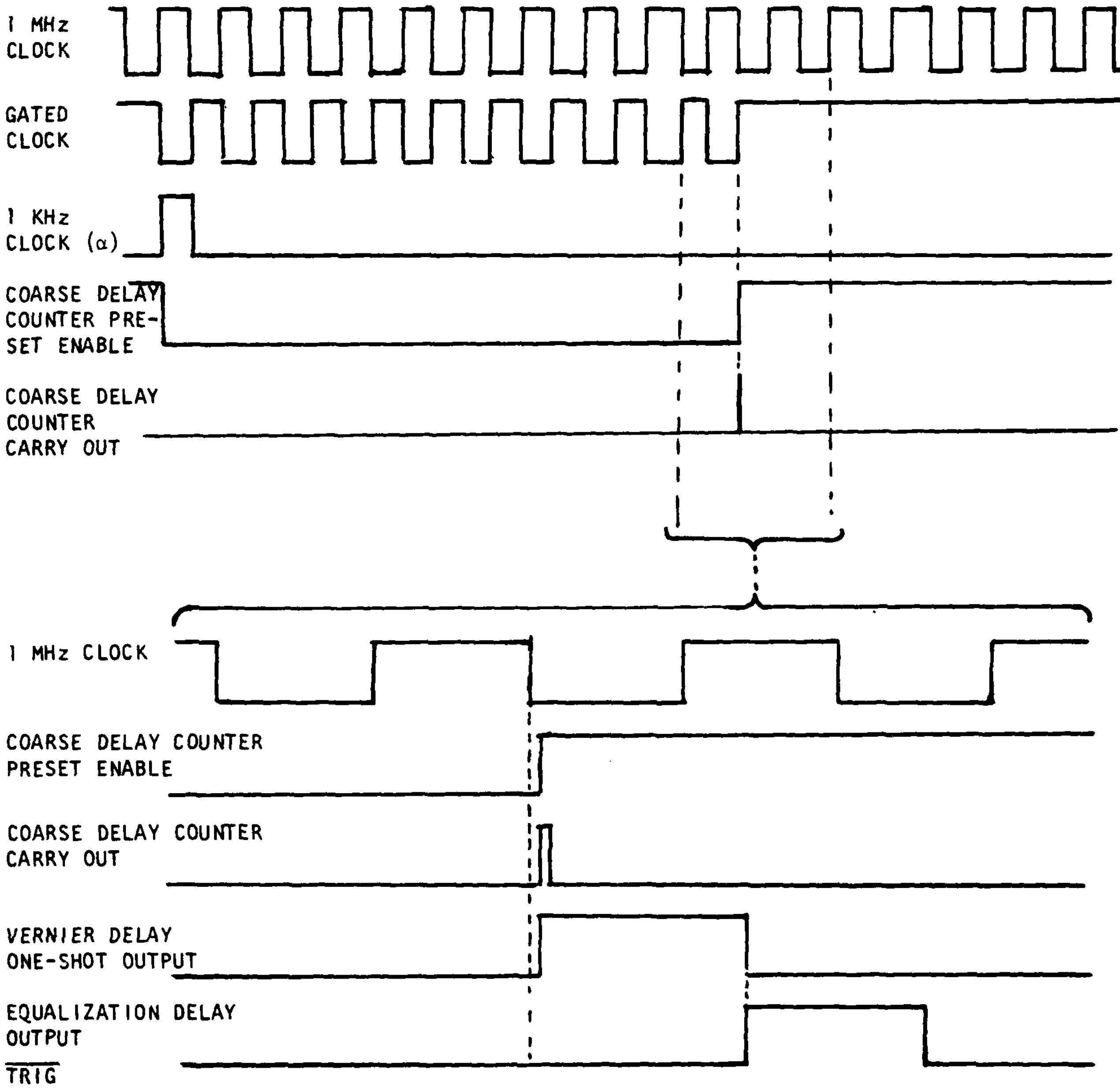


Fig. III-10. Delay board (board #7) timing chart.
Delay setting = 10.7 μs.

and "Single-Step" simply applies the debounced switch output to the counter. The count can be up or down, but will not "wrap around" from 000.0 to 999.9 or vice versa. We inhibit "wraparound" by gating the clock inputs with the carry output of the counter (see Fig. 9). The counter is preset to 000.0 when powered up. The content of the counter is sent to board #9 (the UART board) for transmission to the computer as the "Delay" parameter.

The desired delay is achieved in the system in two parts: coarse delay, and vernier delay. As shown in Fig. 10, prior to the 1 KHz clock pulse, the coarse delay counter is preset enabled, so that the three most significant digits of the delay-set count are loaded into the coarse delay counter. When the 1KHz clock pulse arrives, the preset is disabled, and the 1 MHz clock input is gated on. The counter counts down to 0, at which time the 1 MHz clock is gated off. Then the preset is enabled again. The gate-pulse transition triggers the one-shot which determines the vernier delay. The latter is governed by the least significant digit of the delay-set count. This value controls an analog switch, which control the timing resistance for the vernier delay one-shot. The output of the vernier delay one-shot is further delayed by a fixed amount, to correct for delays introduced by the pulser.

Board #7 also holds the output pulse drivers, which provide 50Ω, 5 V pulses to the pulser/receiver and oscilloscope. The TRIGGER pulse is 2.0 μs wide. Its rising edge triggers the "main bang", and provides oscilloscope sync. The DELAY pulse's rising edge coincides with the first waveform sample; the falling edge comes at a time approximately 0.5 μs after the last waveform sample.

Board #7 ADJUSTMENTS SUMMARY

NAME	EQUIPMENT REQUIRED	PROCEDURE
Vernier Adjust	50 MHz oscilloscope	Using TRIGGER output as sync for oscilloscope, monitor DELAY output. Set time/div. of scope to 0.5 μs/div. Using an insulated adjustment tool, set the Venier adjust capacitor so that each step of the SINGLE STEP delay switch adds 0.1 μs to the DELAY output pulse width.
Delay Equalization	None	Perform ramp delay adjustment on Board #5. Set time/div on oscilloscope monitor to 0.1 μs/div. Set Filter/Digitizer delay to 0, BW controls at 1.5 MHz: Set delay equalization potentiometer so the beginning of the brightened trace coincides with the start of the excitation pulse response.

G. Board #8 (Timing Generators)

Board #8 provides timing and control pulses used by the ADC (Board #4), delay generator (Board #7) and output multiplexer and UART (Board #9). The staircase waveform used in sample-pulse generation (Board #5) is also generated here. Fig. III-11 shows the timing relationships of the various pulses. The staircase generator is a 3-digit BCD ADC (Binary coded decimal analog to digital converter), driven by a 3-digit BCD counter, with a count length of 448.

A POWER-UP pulse is generated a few milliseconds after power is applied to the circuitry. This pulse is used to initialize the delay generator and the UART.

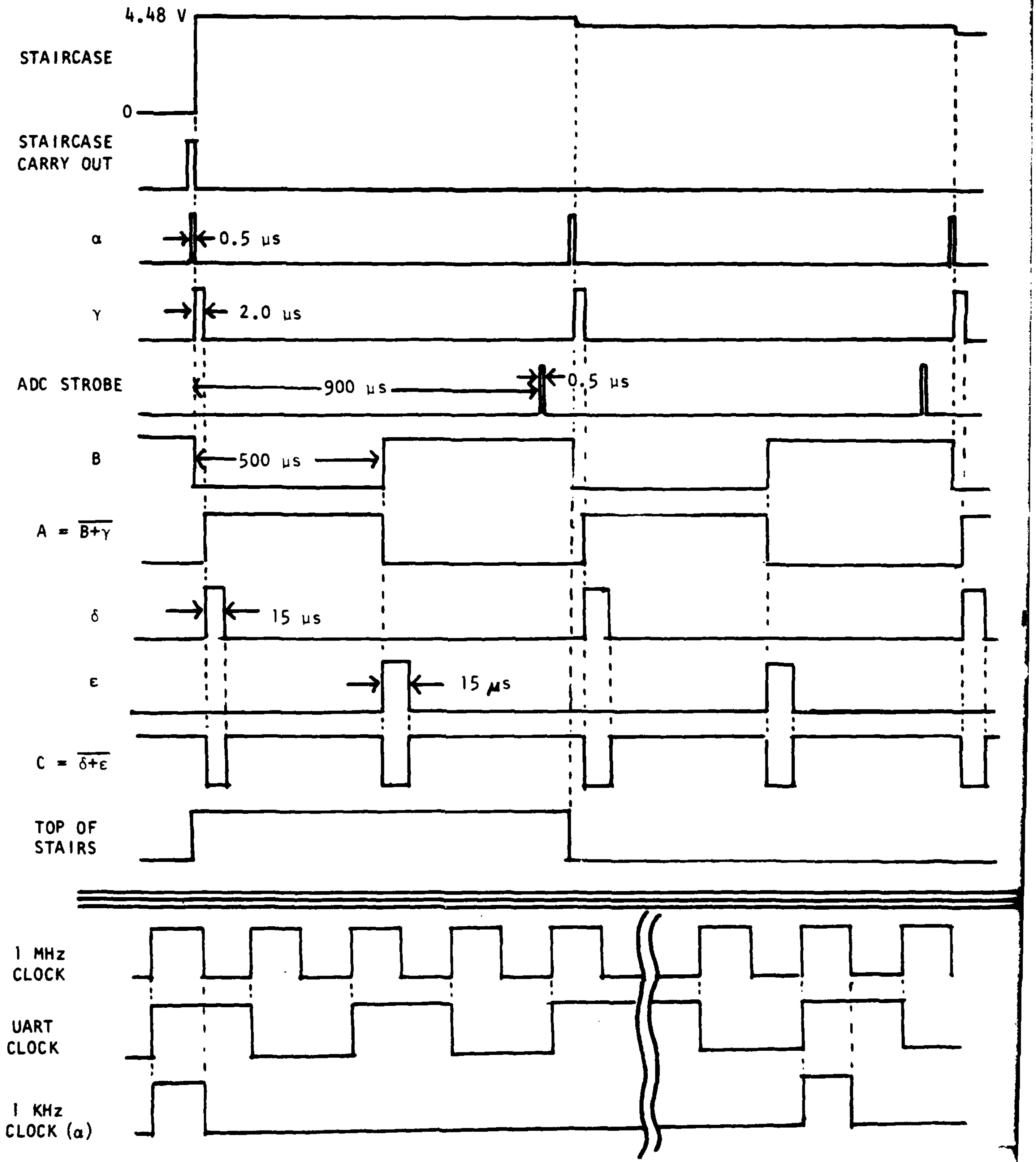


Fig. III-11. Timing generator waveforms (board #8)

H. Board #9 (Output Multiplexer and UART)

The function of Board #9 is to encode and transmit data from the Filter/Digitizer to the computer. Two eight-bit bytes are trasmitted during each millisecond. We will call a pair of bytes one "word." The first byte of a word is the "A" byte, the second, the "B" byte. Two options are available for encoding the snapshot data. In the original format, the eleven most significant bits of each waveform sample are transmitted. The revised format was added to permit more rapid waveform display; in this format only the nine most significant bits of each waveform sample are transmitted. In both formats, the first bit of the first word of each snapshot is set to 1. In every other case, the last bit of every byte is 0. This allows the computer to find quickly the start of the snapshot , and distinguish "A" bytes from "B" bytes. Along with the sample data, each word contains two bits which indicate from which of the three channels the data originated.

All the remaining data, FREQUENCY and BANDWIDTH control settings, TIMEBASE control setting, LOG/LIN setting, and the DELAY setting, are time-multiplexed into the remaining bit (the second) of each word. These data constitute 38 bits. Therefore, they are transmitted within the 38 words representing the first 38 samples in each snapshot. The multiplexing of the parameter data into the transmitted words is controlled by the staircase counter's output. The overall functional block diagram (Fig. B-1) shows how this is accomplished, using tri-state buffers and multiplexers. Note that the first sample of each waveform corresponds to stairstep #448,

and that the staircase counter counts down from that level to zero.

Fig. III-12 illustrates the transmitted or output data formats of the Filter/Digitizer. This serial data sequence is inverted, buffered, and transmitted via an RS232 port to the computer. The signal is again inverted and the order of the data bits reversed before the data is placed in memory, in the format illustrated in Fig. IV-2.

I. Board #10

Board #10 provides power distribution to the other boards of the unit. Connections to the power supply are hardwired into Board #10, through the board connector, to the backplane. Board #10 cannot be removed except by unscrewing the connections at the power supply. Specifications for the power supply are given in Appendices "X" - "Y." The only circuitry mounted on Board #10 itself is the -12 V supply, consisting of a single low-power TO-92 package 3-pin regulator and a few associated capacitors, as shown in Fig. 13. The connections to Board #10 are given in Tables 1 & 2.

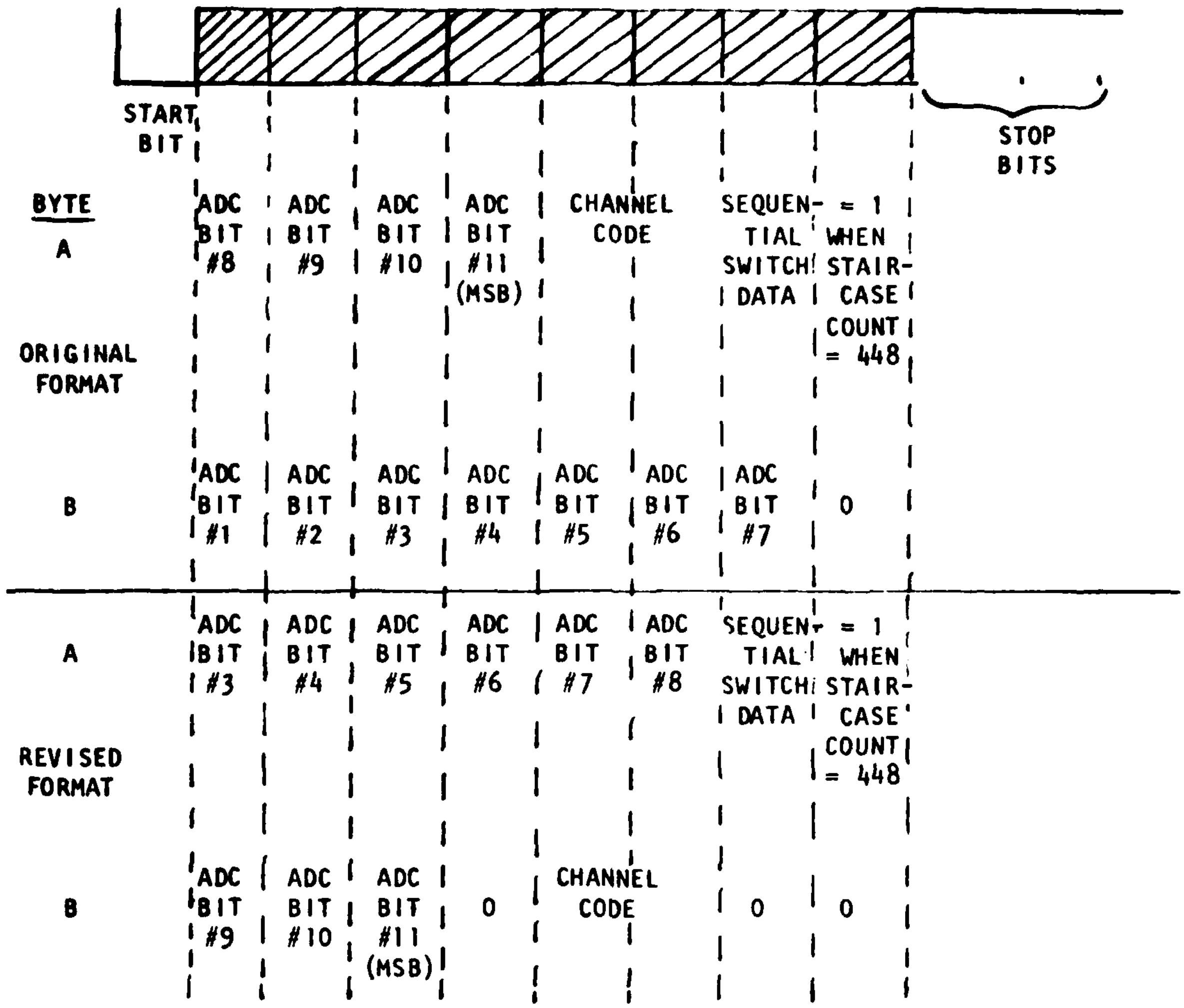


Fig. III-12. UART output data format.

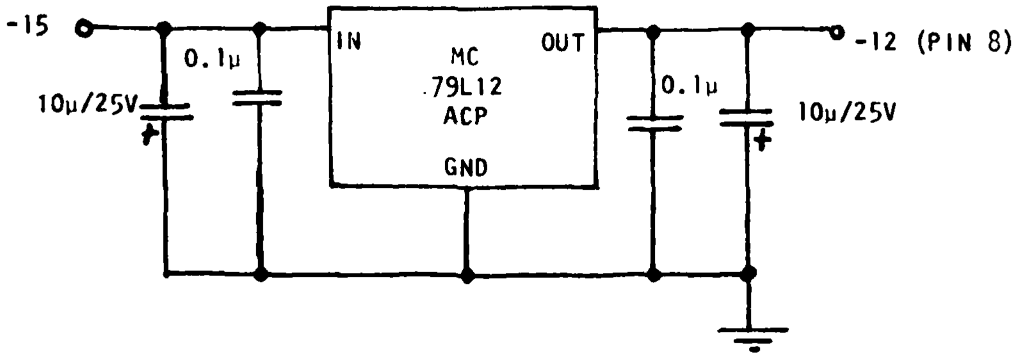


Fig. III-13. -12 V regulator schematic (Board #10).

IV. DESCRIPTION OF OPERATING PROGRAMS

A. Overview

The software which operates the Pulse-Echo Display System comprises five individual, but closely related programs written in BASIC, one short machine-language routine and a Function Key storage file. The operating programs are, for the most part, self cuing, with numerous instruction printouts guiding the operator in making selections of functions, display modes, graphic scale factors, and so on.

All of the programs associated with system operation are stored on a single flexible disk labelled "Operating Programs," which is normally accessed from Disk Drive #1. Most of the Chromatics Disk Operating System (DOS) "resident" commands are also included on the same disk. Furthermore, a number of diagnostic and test programs employed during the development of the system have been left in the repertoire of the operating disk, although there is no further anticipated need for their use until system changes are introduced.

In the program description throughout Section 4 of this report, it is presumed that the reader will have read and be moderately familiar with the descriptive material in the Chromatics 1998 Operating Manual, the accompanying BASIC and DISC System Manuals, and their appended data and instruction sheets. The present discussion will focus on the software which has been specifically developed for the Multiple-Frequency Pulse-Echo Display System.

1. Initial setup

In setting up the programs at the start of an operating session, the operator will retrieve from the disk the key functions stored there. The command sequence:

RESET BOOT F1 DISK OS FETCH KEYFILE.KEY 4000 CR

will reset the computer and then cause the key functions to be copied from the disk into dedicated memory (beginning at HEX 4000). The present system employs key functions 2,3,4,5, and 6. All ensuing activities of the Pulse-Echo System are initiated by means of these 5 keys and the program-cued responses.

2. Summary of the individual programs

a. Program SIMU (in BASIC)

This was devised to compute and place in memory a data sequence which simulates one that might be obtained from the Filter/Digitizer. The program was used extensively in system development before the Filter Digitizer became operational. It may prove useful again when display-related changes are being made in the main operating programs. It permits one to define arbitrary curves for each of the three channel displays.

b. Program START (In BASIC)

This program begins the normal operating sequence. It loads into memory the machine-language routine ("SNAPSHOT") for acquiring data from the Filter/Digitizer, prints a cuing message for the operator and sets a certain sequence-monitoring code, ("flag"), into memory. (Note that the SNAPSHOT program sequence is embodied within START; it is not stored on disk separately.) Program START concludes by loading Program GRAPH into active memory.

c. Program GRAPH (in BASIC)

GRAPH is the display-producing program and is therefore the principal (and most extensive) member of the group.

d. Program GRAFANEX (in BASIC)

This is really an extension to Program GRAPH, stored and used separately however, to conserve active memory space. Its principal function is storing away display data and parameters on a file disk (usually on Drive #2) for later retrieval.

e. Program RETRIEVE (in BASIC)

RETRIEVE serves to assist the operator in locating in disk file a desired, previously-stored data set, and entering this into the system for display and processing by Program GRAPH. RETRIEVE ends by loading GRAPH into active memory and running it.

f. Program KEYFILE

This short program serves to define the actions of Function Keys 2 through 8. (7 and 8 are not used, however, in the present system.) It is called into active memory during initiation of operation, as indicated in paragraph IV-A-1.

3. Actions of the Function Keys

a. Function Key #1

This key is used when defining other key functions or in calling a disk KEYFILE into active memory. (See Chromatics Manual.) It is not an "operating" function key.

b. Function Key #2

With this key, we set up the BASIC interpreter, call Program SIMU from disk memory, and run it. When SIMU ends, another key must be actuated for continued activities on the system.

c. Function Key #3

This key serves to set up BASIC (as does Key #2) and then load and run Program START. It is usually the first key used in an operating session.

d. Function Key #4

Key #4 presumes that Key #3 has been activated first. (Otherwise an error will be indicated). It runs GRAPH in the "SNAPSHOT" mode.

e. Function Key #5

This key presumes that Key #2 or #3 has been activated previously, so that BASIC has been called up. Key #5 serves to activate the RETRIEVE program which is used in loading data from an existing file on a disk. As noted in IV-A-2 e, GRAPH succeeds RETRIEVE automatically when the latter is complete.

f. Function Key #6

In similar fashion to #5, this key also presumes that BASIC has been set up by Key #2 or #3, (or otherwise). Key #6 runs Program GRAPH using data that is already stored (perhaps from the just-previous processing operation) in active memory.

B. Memory Allocations

The Pulse-Echo Display System uses four areas of memory in the Chromatics 1998 Computer for its special-purpose programs:

1. The dedicated Function Key storage area:

Beginning at location DEC 16384 (DEC means the number is expressed in decimal, as opposed to hexadecimal form), 64 places (bytes) are reserved for each of Keys #2 through 8. Appendix C-1 is a short BASIC program, named KEE, written to display the contents of the key function memory area. This simple program probably will not conveniently display every terminal key code that might be included in a Key Function, but it does serve the present system adequately.

2. Machine language "SNAPSHOT" program

This sequence of 46 steps is stored in scratchpad memory at locations DEC 17000 through 17045, (HEX 4268-4295). The memory area is loaded by Program START.

3. The system program (BASIC) area

This is the memory area beginning at DEC 17344 where the system operating programs are loaded in response to "LOAD" commands to the disk system. When we set up BASIC, we allocate space in this area up to (but not including) DEC 43000 for such programs. It is necessary, therefore, to insure that any single program does not exceed this space. If it does, an error will be indicated and the program will halt.

4. The "buffer" memory area

This is the space "above" the BASIC program area, starting at DEC 43000 (which we have selected for convenience in the present system) and ending at 48475. Most of this memory space has been assigned to specific system parameters, data, and sequence codes or "flags." Table IV-1 summarizes the various items stored in the "buffer" space. (To review the particular format used for storage of the parameters, refer to the Program GRAFANEX listing in Appendix C-6).

The start of the buffer space is referenced by the symbol "IØ" in Program SIMU and by "DA" in the other BASIC programs. BASIC is called up either by Function Key #2 or #3, and in this process the start of the buffer space is defined (at 43000 in the present system). Memory locations 15174 and 15175 are dedicated (by the Chromatics 1998 operating system) to the representation of the number which defines the start of the buffer space. Each of the system programs, therefore, retrieves this location by assignment of DA (or IØ) in the following BASIC statement:

```
DA = 256 * PEEK(15175) + PEEK(15174).
```

By this means, one establishes the buffer start location just once, when BASIC is called up originally, and can be confident that the operating programs will adjust to this definition automatically.

Actually, the number 43000 is a reasonable compromise for the present system; significant changes, up or down, may not be desirable. Sufficient space must be left below the buffer start for the largest BASIC operating program involved, and above the buffer start location for the various data items in Table IV-1.

C. Program Organization

Figure IV-1. illustrates the major sequential features of the operating programs and the Key Functions. We assume the stored key functions have been loaded (as in section IV-A-1). Then, one starts with either F2 or F3. After this, all keys and selections are cued by the respective programs.

Function Keys #4,5,6 presume that BASIC has been called up (by Keys F#2,3 or otherwise), but even here, appropriate error messages will be provided if prerequisite conditions are not fulfilled.

D. Individual Program Features

1. Function Key Programs

Table IV-2 lists the actions called out by the function keys. The table is a printout which results directly from exercising Program KEE, listed in Appendix C-1. Note that "(ESC)B" refers to the escape code for the BASIC interpreter which is initiated on the Chromatics 1998 keyboard by the single key marked BASIC.

TABLE IV-1

USE OF "BUFFER" MEMORY SPACE

LOCATION	DATA STORED	
DA		
+1 to +7	File #	
+8	$1/A_1$	channel linear gain factors
+9	$1/A_2$	channel linear gain factors
+10	$1/A_3$	channel linear gain factors
+11	$8 \cdot T$	(scale factor)
+12		
+13	$4 \cdot f_1$	channel frequencies
+14	$4 \cdot f_2$	channel frequencies
+15	$4 \cdot f_3$	channel frequencies
+16	$4 \cdot B_1$	channel bandwidths
+17	$4 \cdot B_2$	channel bandwidths
+18	$4 \cdot B_3$	channel bandwidths
+19	"4," "5," "6" key flags	
+20	"1" = file flag	
+21	"1" = one-time flag (message is presented prior to full plot once per session)	
+22	time-base duration code	
+23 (High order digit) to +26	Delay (before time base begins)	
+27	lin/log code	
+28	D_1	channel logarithmic gain offsets
+29	D_2	channel logarithmic gain offsets
+30	D_3	channel logarithmic gain offsets
+100 to +5475	5376 data bytes	

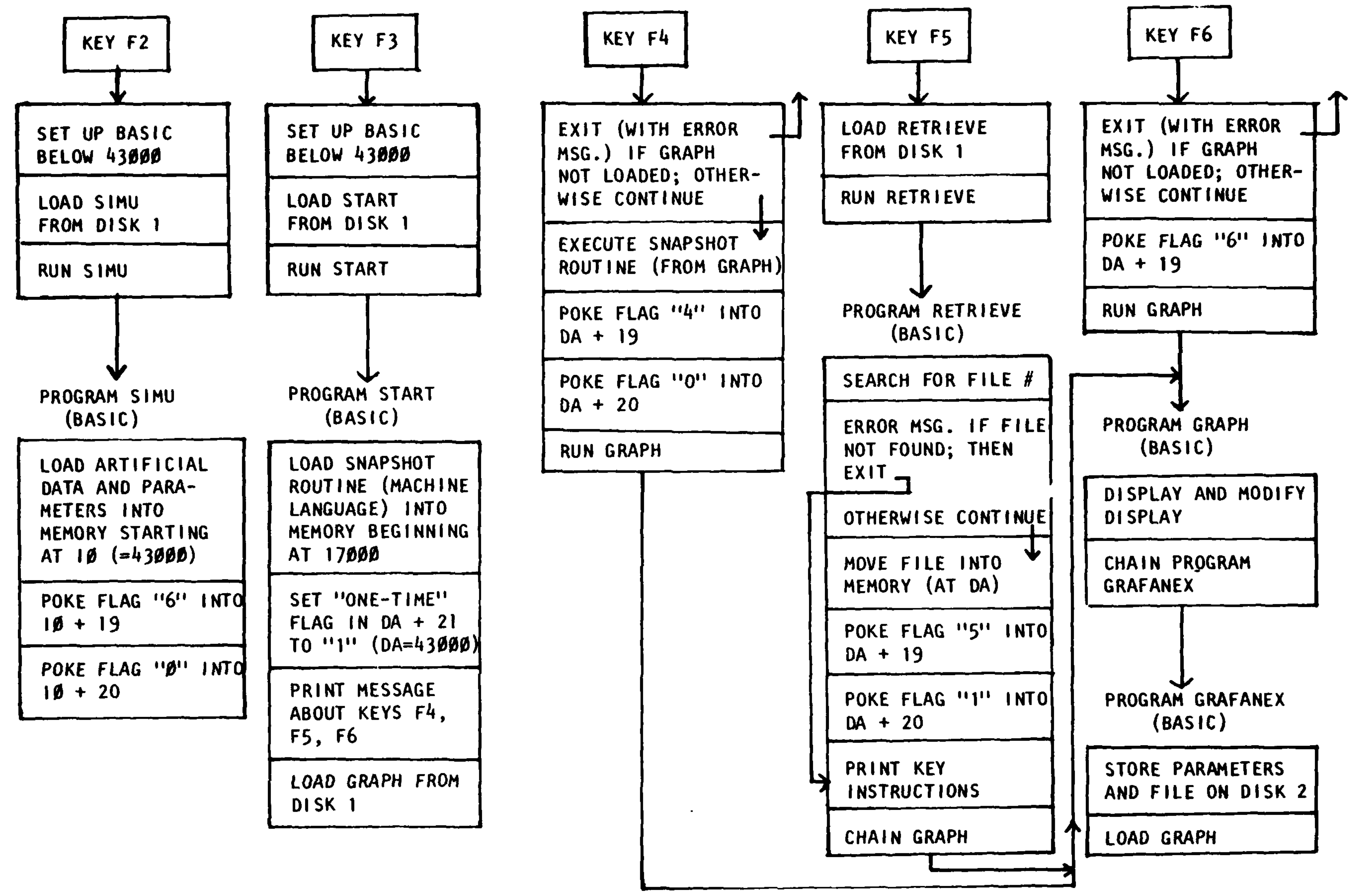


Fig. IV-1. Summary of overall program organization.

TABLE IV-2

FUNCTION KEY ACTIONS

(USING KEE)

```
KEY F 2
(ESC)B43000
DOS"LOAD/1 SIMU/1"
RUN

KEY F 3
(ESC)B43000
DOS"LOAD/1 START/1"
RUN

KEY F 4
(ERASE)(MODE)C4SNAPSHOT not loaded; press (MODE)C5F3 (MODE)C4first.
(MODE)CO
RUN 9000

KEY F 5
DOS"LOAD/1 RETRIEVE/1"
RUN

KEY F 6
(ERASE)(MODE)C4GRAPH not loaded; press key (MODE)C5F3 (MODE)C4first.
(MODE)CO
RUN 8000
```

2. Program SIMU

This data-simulation program is listed in Appendix C-2. The three channel response curves are defined in *SIMU*, in steps 470, 480, and 490. Based on a full-scale amplitude of (2"-1) = 2047 vertical units, the three linear-mode curves are defined as follows:

$$y_1 = 340 \sin \left(\frac{N + 64}{40} \right) + 400$$

$$y_2 = 800 \cos \left(\frac{N}{80} \right) + 800$$

$$y_3 = 370 - \frac{6(N - 200)^2}{1000} ,$$

where N is the horizontal axis coordinate running from 0 to 447. Logarithmic versions of the curves are generated in program steps 510, 520, 530. One can easily change these waveforms to simulate other desired curves. Table IV-3 indicates the display parameters simulated by SIMU.

3. SNAPSHOT Routine

This sequence of 46 machine-language steps is loaded into memory by Program START (steps 150 - 200), and executed by Program GRAPH (steps 9020 - 9110). Refer to the listing in Table IV-4.

In steps (relative locations) 0 through 6, Register HL is loaded with a number representing the starting location of the memory field that will hold the snapshot data. This location is 100 bytes (decimal beyond the start of the create buffer.

TABLE IV-3

PARAMETERS STORED IN MEMORY BY PROGRAM SIMU

	CHAN 1 (GREEN)	CHAN 2 (RED)	CHAN 3 (CYAN)
CENTER FREQUENCY	1.0 MHz	3.0 MHz	9.0 MHz
BANDWIDTH	.25 MHz	.5 MHz	1.0 MHz
DELAY		325.4 μseconds	
TIME BASE		10 μseconds	

DATA BIT SEQUENCE

S(0)	0	Delay	S(12)	0	Delay	S(25)	1	
S(1)	0	(hundreds	S(13)	1	(tenths	S(26)	1	f_3
S(2)	1	digit)	S(14)	0	digit)	S(27)	0	
S(3)	1		S(15)	0		S(28)	0	
S(4)	0	Delay	S(16)	0	Time	S(29)	0	B_1
S(5)	0	(tens	S(17)	1	base	S(30)	0	
S(6)	1	digit)	S(18)	1	duration	S(31)	1	B_2
S(7)	0		S(19)	0		S(32)	1	
S(8)	0	Delay	S(20)	0	f_1	S(33)	1	B_3
S(9)	1	(unit	S(21)	1		S(34)	0 = linear	
S(10)	0	digit)	S(22)	0			1 = logarithmic	
S(11)	1		S(23)	1	f_2	S(35)	0 Channel 1	WIDEBAND SWI
			S(24)	1		S(36)	0 Channel 2	SETTING
						S(37)	0 Channel 3	

SEQUENCE FLAGS

LOCATION	VALUE	SIGNIFICANCE
DA + 19	6	Stored SIMU data can be called into GRAPH by Function Key #6
DA + 20	0	Data does not originate from a stored file on disk.

TABLE IV-4

MACHINE LANGUAGE "SNAPSHOT" ROUTINE

RELATIVE LOCATION	ABSOLUTE LOCATION DECIMAL	ABSOLUTE LOCATION HEX	OP CODE (HEX)	DESCRIPTION
Ø	17ØØØ	4268	2A	LOAD(HL) with contents of (15174),(15175);
1	17ØØ1	4269	46	(beginning location of Create Buffer.)
2	17ØØ2	426A	3B	
3	17ØØ3	426B	Ø1	LOAD(BC) with DECIMAL 1ØØ = Hex 64
4	17ØØ4	426C	64	(offset from Create Buffer start)
5	17ØØ5	426D	ØØ	
6	17ØØ6	426E	Ø9	ADD contents of (BC) to those of (HL). (This is 100 beyond the start of the buffer; the data begins here.)
7	17ØØ7	426F	Ø1	LOAD (BC) with DECIMAL 5376 = HEX 15ØØ
8	17ØØ8	427Ø	ØØ	(total count of bytes to be loaded into buffer.)
9	17ØØ9	4271	15	
1Ø	17Ø1Ø	4272	1E	LOAD (E) with DECIMAL 5Ø = HEX 32 (arbitrary
11	17Ø11	4273	32	number of trials before an error is declared.)
12	17Ø12	4274	DB	LOAD UART Receiver Status into (ACC)
13	17Ø13	4275	4D	(8 bits)
14	17Ø14	4276	CB	TEST BIT 1 OF (ACC)
15	17Ø15	4277	4F	(Receiver Ready)
16	17Ø16	4278	CA	IF ZERO FLAG = 1 (receiver <u>not</u> ready)
17	17Ø17	4279	8B	THEN JUMP TO HEX LOCATION 428B;
18	17Ø18	427A	42	(relative location 35.)
19	17Ø19	427B	DB	LOAD UART DATA INTO (ACC)
2Ø	17Ø2Ø	427C	4C	(8 bits)
21	17Ø21	427D	77	LOAD contents of (ACC) into location defined by (HL). (This is where it will be picked up by BASIC Program GRAPH.)
22	17Ø22	427E	23	INCREMENT (HL)
23	17Ø23	427F	ØB	DECREMENT (BC)
24	17Ø24	428Ø	3E	LOAD Ø into (ACC)
25	17Ø25	4281	ØØ	
26	17Ø26	4282	B8	COMPARE contents of (A) and (B). (Higher level bits in BC.)
27	17Ø27	4283	C2	IF ZERO FLAG = Ø (not at end yet)
28	17Ø28	4284	72	THEN JUMP TO HEX LOCATION 4272;
29	17Ø29	4285	42	(relative location 1Ø.)

TABLE IV-4 (CONTINUED)

RELATIVE LOCATION	ABSOLUTE LOCATION DECIMAL	HEX	OP CODE	DESCRIPTION
3Ø	17Ø3Ø	4286	B9	COMPARE contents of (A) and (C). (Lower bits in BC.)
31	17Ø31	4287	C2	IF ZERO FLAG = Ø (not at end yet)
32	17Ø32	4288	72	THEN JUMP TO HEX LOCATION 4272;
33	17Ø33	4289	42	(relative location 1Ø.)
34	17Ø34	428A	C9	RETURN
35	17Ø35	428B	1D	DECREMENT (E).
36	17Ø36	428C	3E	LOAD Ø into (ACC).
37	17Ø37	428D	ØØ	
38	17Ø38	428E	BB	COMPARE contents of (A) and (E).
39	17Ø39	428F	C2	If ZERO FLAG = Ø (not at end yet)
4Ø	17Ø4Ø	429Ø	74	THEN JUMP TO HEX LOCATION 4274;
41	17Ø41	4291	42	(relative location 12.)
42	17Ø42	4292	32	LOAD contents of (A) (now Ø) into HEX location 42CC; (decimal location 171ØØ; this is where BASIC Program GRAPH will pick this indicator of "bad" or missing data. See after GRAPH step #9ØØØ.
43	17Ø43	4293	CC	
44	17Ø44	4294	42	
45	17Ø45	4295	C9	RETURN

In steps 7 through 9, Register BC is then loaded with the number of snapshot bytes that will be taken in. (This is 5376 - 448 points per channel x 2 bytes per point x 3 channels x 2 complete sets of data.) In steps 10, 11, Register E is loaded with the number 50, which is the maximum of sequential, unsuccessful attempts to look for incoming data that will be allowed before a fault is declared.

In steps 12 through 18, the terminal's UART "receiver ready" status is tested: if not ready, we jump to step 35; if ready, we continue with steps 19 - 21, which load the UART data (8 bits) into the location defined by the contents of Register HL. When the data has been loaded, we increment Register HL by 1, step 22, (getting ready for the next byte loading) and decrement Register BC, step 23, *crossing off, so to speak*, another one of the 5376 data bytes expected.

Note that two UARTs are used in the Pulse-Echo Display System. One, the "transmitting UART is located in the Filter/Digitizer. The other "receiving" UART is located in the Chromatics terminal. In the description of the operating programs, the UART referred to is always the "receiver." In the description of the Filter/Digitizer, the UART referred to is always the "transmitter."

In steps 24 through 29, we look for all zeros in Register B. (These are the higher-weight bits in combined Register BC.) Similarly, in steps 30 - 33, we examine the lower-weight bits in Register C for all zeros. If both registers do not contain all zeros

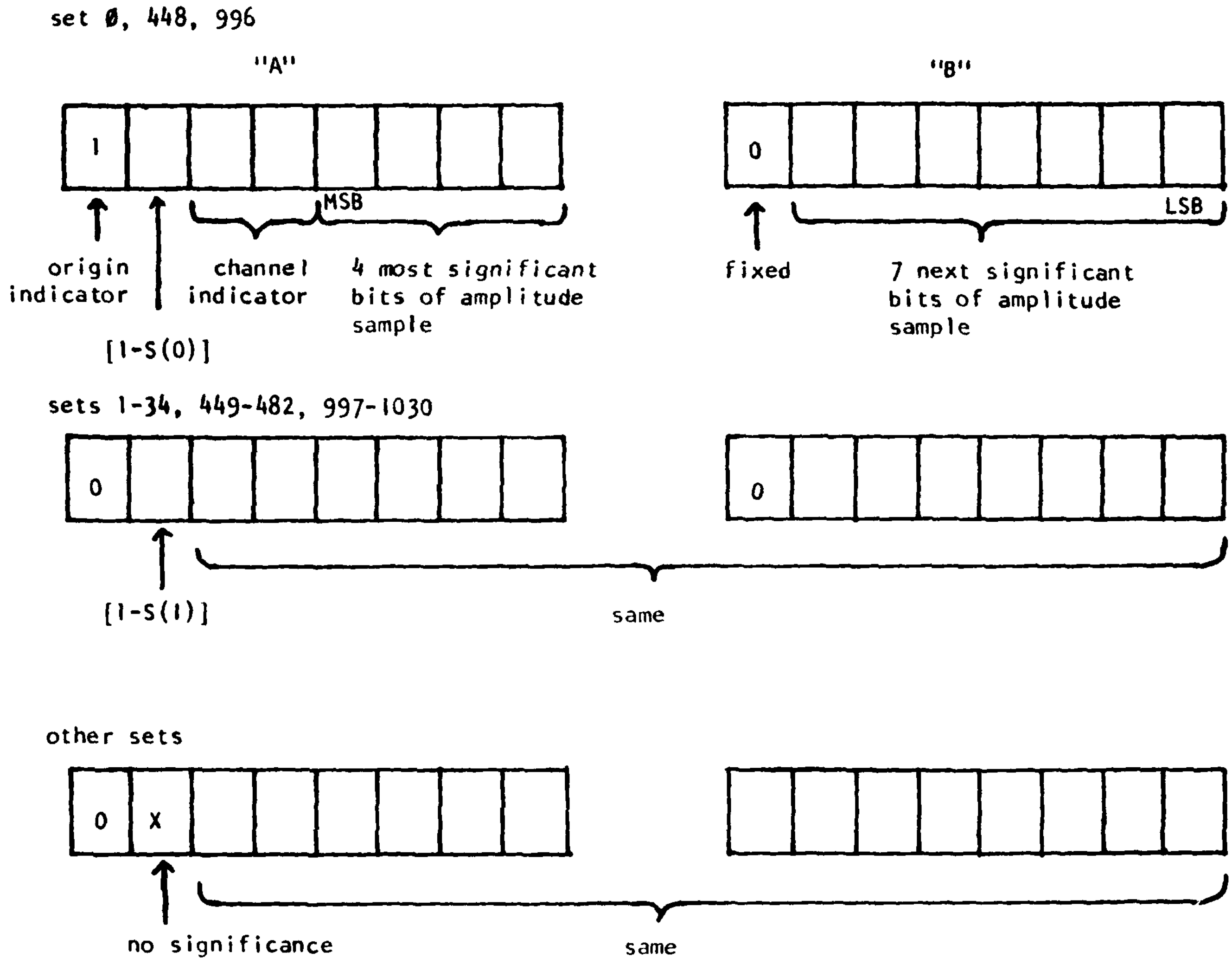
(meaning that all 5376 bytes have not yet been acquired), we return to step 10 and repeat the byte-acquisition process. If Register BC (both parts) contains all (i.e., 16) zeros, then step 34 returns to the BASIC program which originally called up the machine-language routine.

Recall that in steps 12 through 18, if the UART proved not to be "ready," we jumped to step 35. In this, we decrement Register E, beginning or continuing the countdown from 50. In steps 30 - 41, we test Register E for zero. If the test fails, we jump back to look for the next UART "ready" flag. If, however, the test shows that we have reached 50 unsuccessful attempts at finding data at the UART receiver, we "give up." In steps 42 - 44 the number 0 is loaded into the memory location 17100 which will signify to the BASIC Program GRAPH, (when it looks there), that the snapshot data link is somehow defective. Step 45 returns control to BASIC.

The Filter/Digitizer output is a continuously-repeating sequence of 2688 bytes, each of 8 bits. SNAPSHOT records two cycles of the sequence in buffer space. The "phase" (start point) of the recorded cycles depends on the instant the SNAPSHOT command is initiated and is thus uncontrolled. When the data is used for display, a search is made to locate a starting point of the sequence. Each sequence comprises 1344 sets of alternate "A" and "B" bytes. The sets have the format shown in Fig. IV-2.

4. Program START

This program, which sets up the SNAPSHOT sequence, is listed in Appendix C-3.



Channel # and "S" bits are defined in Table IV-5

Fig. IV-2(a). Data sequence produced by snapshot routine, using original (11 bit) data format.

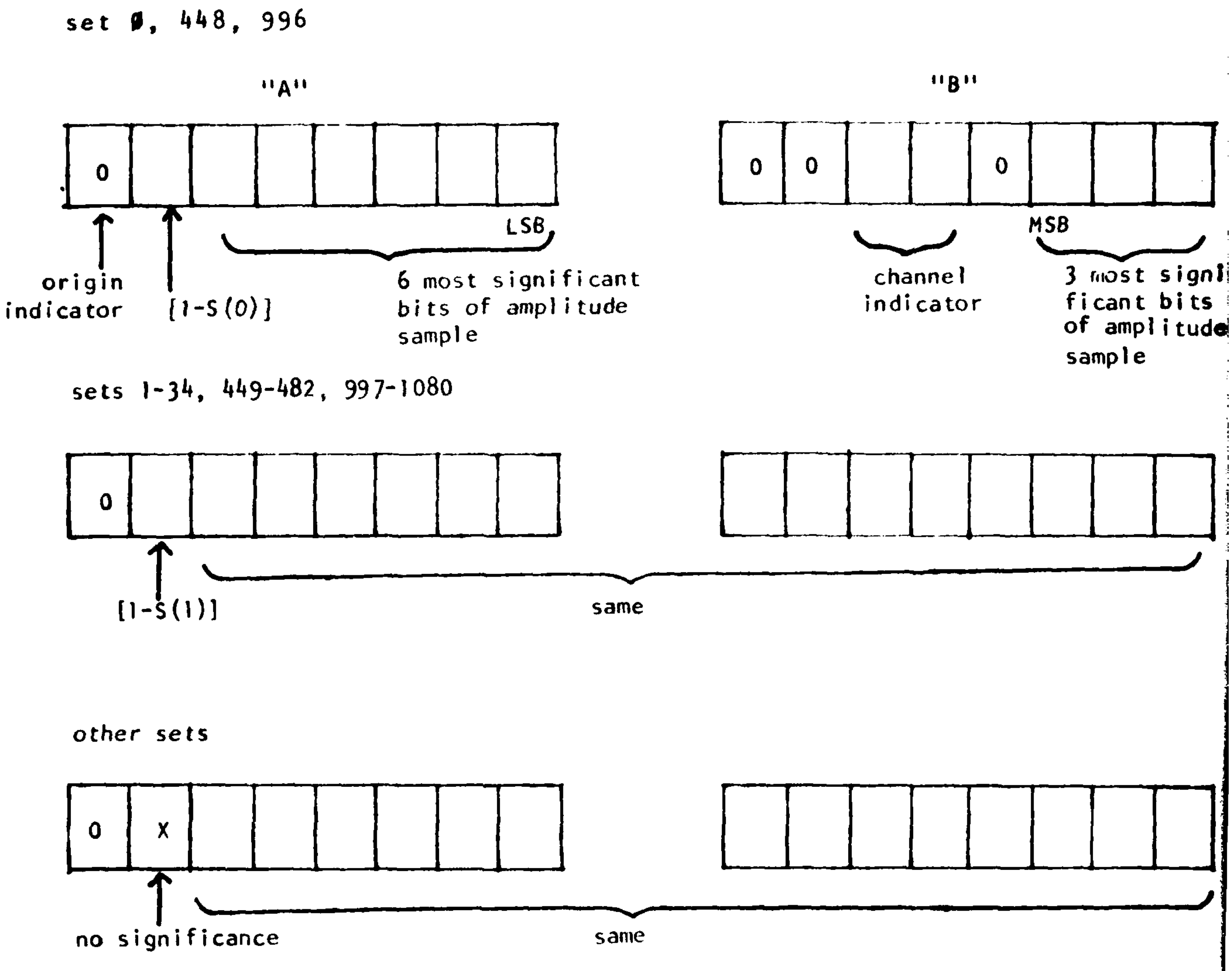


Fig. IV-2(b). Data sequence produced by snapshot routine, using revised (9 bit) data format.

TABLE IV-5

CHANNEL CODES AND S(I) BITS USED IN THE SNAPSHOT DATA SEQUENCE

CHANNEL CODES:

BIT CODE	CHANNEL #	COLOR
00	1	Green
01	2	Red
10	3	Cyan

PARAMETER BITS [S(I)]:

I	PARAMETER REPRESENTED BY BIT CODE S(I)	I	PARAMETER REPRESENTED BY BIT CODE S(I)	I	PARAMETER REPRESENTED BY BIT CODE S(I)
0	DELAY 100 microseconds (BCD)	12	DELAY 0.1 microseconds (BCD)	25	f_3 code
1		13		26	
2		14		27	
3		15		28	B_1 code
4	DELAY 10 microseconds (BCD)	16	Time base code	29	
5		17		30	B_2 code
6		18		31	
7		19	f_1 code	32	B_3 code
8	DELAY microseconds (BCD)	20		33	
9		21		34	LIN/LOG code
10		22	f_2 code		
11		23			
		24			

Code significance of the parameter bits is defined on next page

TABLE IV-5 (CONTINUED)

CODE SIGNIFICANCE OF THE FILTER-DIGITIZER PARAMETER BITS

CHANNEL CENTER FREQUENCIES (MHz)

CHANNEL CODE	000	001	010	011	100	101	110	111
1	.75	1.0	1.25	1.5	1.75	2.0	2.25	2.5
2	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5
3	3	4	5	6	7	8	9	10

ALL BANDWIDTHS (B_1, B_2, B_3)

CODE	00	01	10	11
MHz	.25	.5	1	1.5

TIME BASE DURATION

CODE	000	001	010	001	100	101
μSEC	1	2	5	10	20	50

LIN/LOG CODE: 0 = Linear Display 1 = Logarithmic Display

5. Program RETRIEVE

RETRIEVE is used for searching a disc and loading a desired file into the buffer memory. It is listed in Appendix C-4.

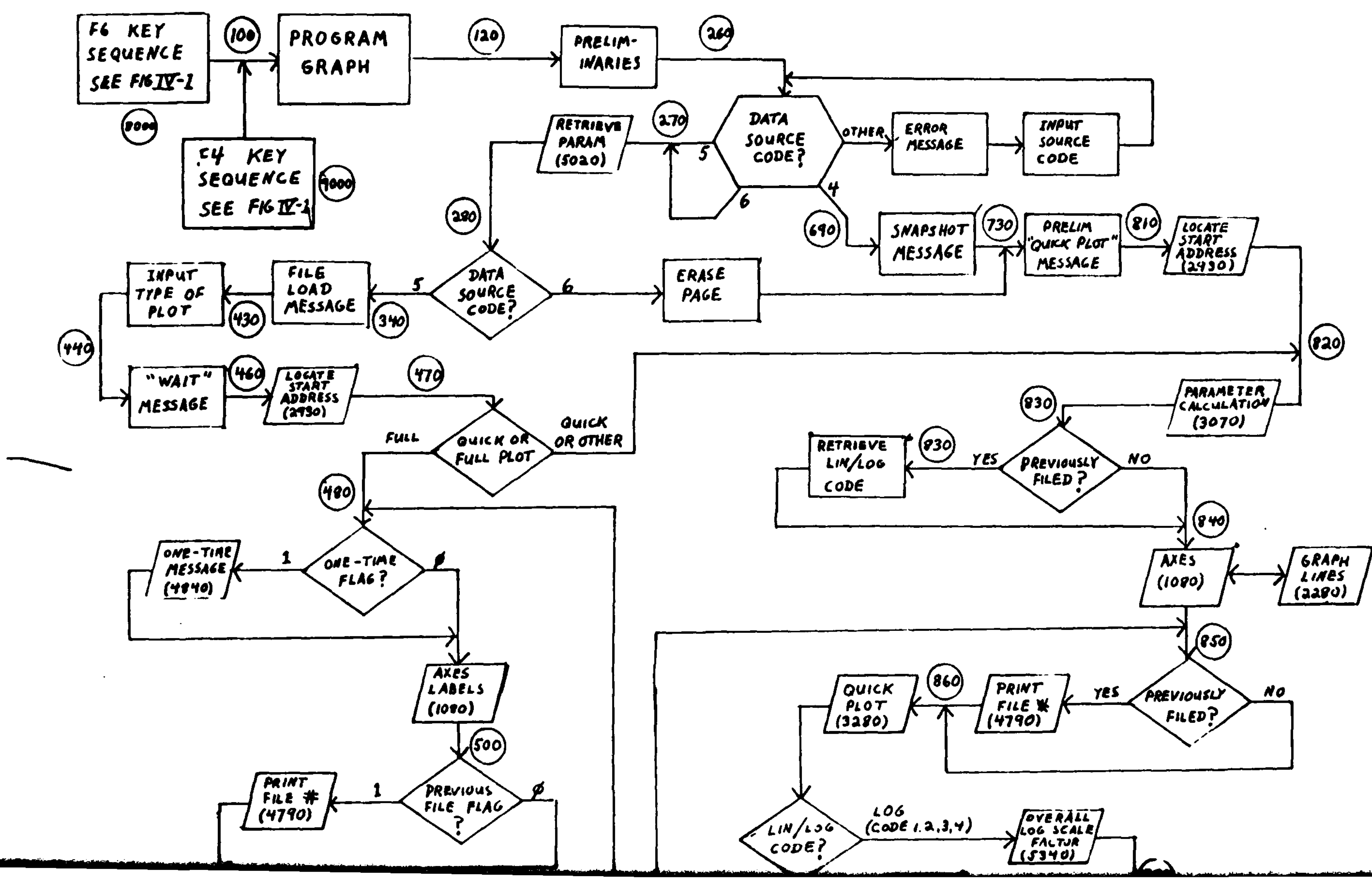
6. Programs GRAPH AND GRAFANEX

As noted, these two are really sections of one program. The separation has been made to limit the size of the program memory space, so that the buffer space remaining would be large enough to hold the display data required. When the final data store-away routines are required, GRAPH "chains" to GRAFANEX. At the end of the latter, GRAPH is reloaded.

Figure IV-3 is a flow chart summarizing the structure and functions of GRAPH and GRAFANEX. The circled numbers represent step numbers in the program listings in Appendices C-5 and C-6, respectively.

For reference purposes, Table IV-6 presents an index of the various subroutines employed in the Programs GRAPH and GRAFANEX. An index to all symbols employed in the two programs is also given for reference, in Table IV-7.

Figure IV-4 and IV-5 are photographs of typical displays produced during exercise of Program GRAPH. Figure IV-4 shows a 3-channel Full Plot, along with display of relevant channel parameters. Figure IV-5 illustrates a message and the information displayed when processing of a particular data set has been completed.


F6 KEY SEQUENCE SEE FIG IV-1
8000
100
PROGRAM GRAPH
F4 KEY SEQUENCE SEE FIG IV-2
9000
120
PRELIM-INARIES
260
DATA SOURCE CODE?
OTHER
ERROR MESSAGE
INPUT SOURCE CODE
270
RETRIEVE PARAM (5020)
5
6
4
690
SNAPSHOT MESSAGE
730
PRELIM "QUICK PLOT" MESSAGE
810
LOCATE START ADDRESS (2430)
280
DATA SOURCE CODE?
ERASE PAGE
FILE LOAD MESSAGE
340
INPUT TYPE OF PLOT
430
440
"WAIT" MESSAGE
460
LOCATE START ADDRESS (2430)
470
QUICK OR FULL PLOT
FULL
QUICK OR OTHER
820
PARAMETER CALCULATION (3070)
830
PREVIOUSLY FILED?
YES
NO
RETRIEVE LIN/LOG CODE
830
840
AXES (1080)
GRAPH LINES (2280)
480
ONE-TIME FLAG?
1
Ø
ONE-TIME MESSAGE (4840)
AXES LABELS (1080)
500
PREVIOUS FILE FLAG ?
PRINT FILE # (4790)
850
PREVIOUSLY FILED?
PRINT FILE # (4790)
860
QUICK PLOT (3280)
LIN/LOG CODE?
LOG (CODE 1,2,3,4)
OVERALL LOG SCALE FACTOR (5340)

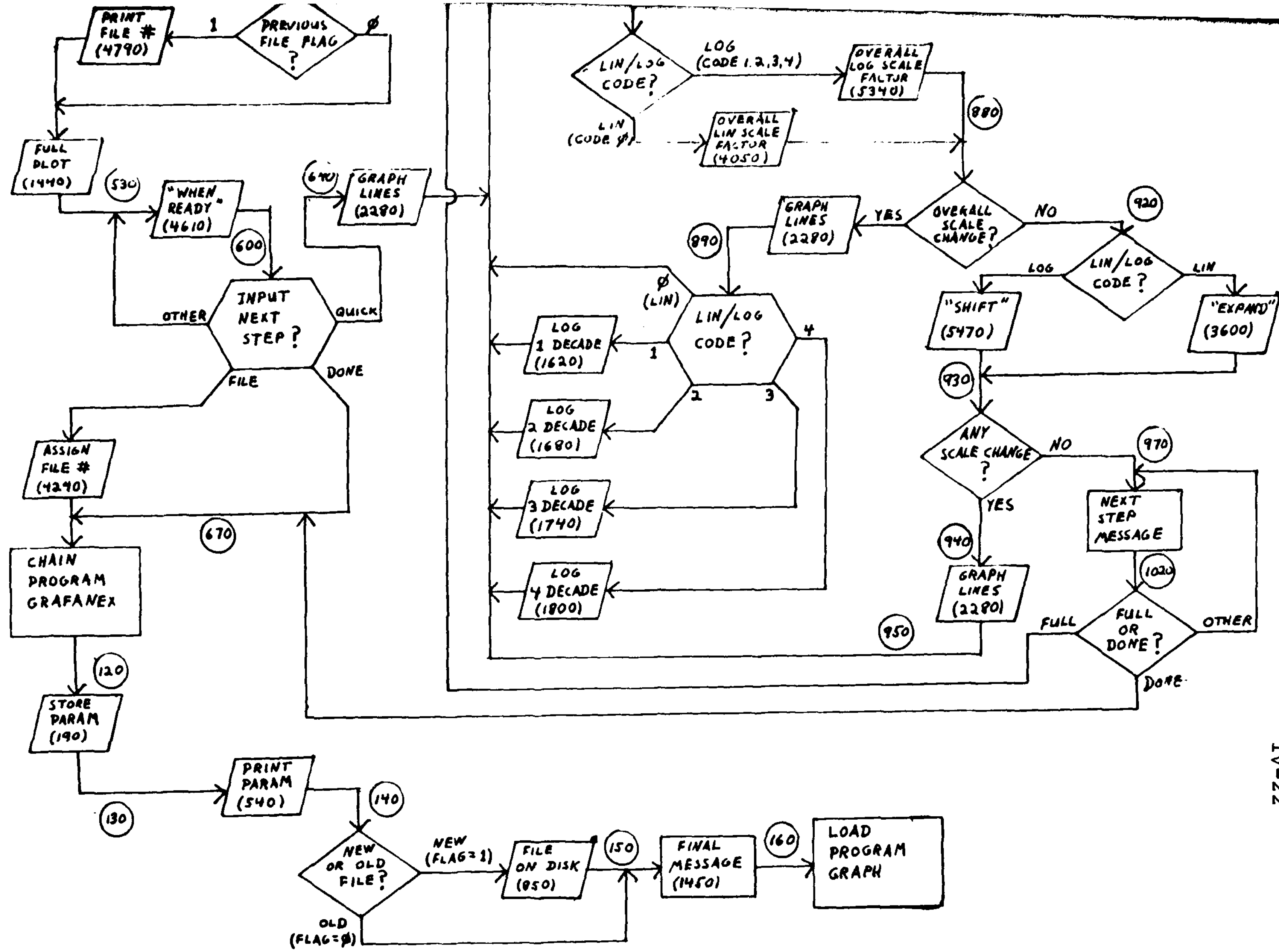


Fig. IV-3. Flow chart for GRAPH and GRAFANEX.

TABLE IV-6

PROGRAM GRAPH SUBROUTINE INDEX

SUBROUTINE	LINE
AXES	1080
VERTICAL AXIS LINEAR	1570
VERTICAL AXIS LOG 1-DECADE	1620
VERTICAL AXIS LOG 2-DECADE	1680
VERTICAL AXIS LOG 3-DECADE	1740
VERTICAL AXIS LOG 4-DECADE	1800
ERASE VERTICAL SCALE	1850
FULL PLOT	1940
GRAPH LINES	2280
IVL = 0 (horizontal scale for interval = 1 μsec)	2490
IVL = 1 (2 μsec)	2570
IVL = 2 (5 μsec)	2650
IVL = 3 (10 μsec)	2720
IVL = 4 (20 μsec)	2790
IVL = 5 (50 μsec)	2860
START ADDRESS SEARCH THRU DATA IN BUFFER	2930
PARAMETER CALCULATION	3070
QUICK PLOT	3280
RESCALING (linear)	3600
SCALE EXP. ENTRY	3940
OVERALL SCALE FACTOR (linear)	4050
ASSIGN FILE #	4240
SAY WHEN READY	4610
PRINT FILE #	4790
MESSAGE PRIOR TO FULL PLOT	4840
RETRIEVE STORED PARAMETERS	5020
LOG RANGE CHANGE	5340
LOG WAVEFORM SHIFT	5470
EXCESS SHIFT MSG	5830

TABLE IV-6 (CONTINUED)

SPECIAL KEY SEQUENCES	LINE
F6	8000
F4	9000

PROGRAM GRAFANEX

ROUTINE	LINE
STORE AWAY PARAMETERS	190
PRINT OUT PARAMETERS	540
FILE ON DISK	850
CALCULATION OF AVAILABLE DISK SPACE	1350
FINAL MESSAGE	1450

TABLE IV-7

VARIABLE INDEX

PROGRAM GRAPH/GRAFANEX

*Note: "X" before a line number refers to program GRAFANEX

VARIABLE	LINE FIRST USED	VALUES ASSIGN-ED AT LINES	MEANING
AA	9050		dummy variable in expression terminating machine language program
AF	X1390	X1390, X1410	available file spaces on disk in Drive #2
AS	X1380	X1380	no. of available sectors on disk in Drive #2
A1	160	160, 3690, 3840, 5060	attenuation to be applied to green curve (linear mode)
A2	160	160, 3750 3840, 5080	attenuation to be applied to red curve (linear mode)
A3	160	160, 3820 3840, 5100	attenuation to be applied to cyan curve (linear mode)
B1	160	160, 3230, 5170	Bandwidth of green channel in MHz
B2	160	160, 3240, 5180	Bandwidth of red channel in MHz
B3	160	160, 3250, 5190	Bandwidth of cyan channel in MHz
CH	2020	2020, 3340	(Integer) Channel No. 0,(green); 1,(red); 2,(cyan)
DA	230	230	Starting location of buffer memory
DEL	160	160, 3130, 3160, 5210, X760	Waveform starting point (delay) in use.
DD	5200	5200, X750	Integer part of DEL
D1	5130	5130, 5570, X640	Amplitude shift (dB) to be added to green waveform (logarithmic mode)
D2	5130	5130, 5650, X650	Amplitude shift (dB) to be added to red wavefrom (logarithmic mode)
D3	5130	5130, 5730, X660	Amplitude shift (dB) to be added to cyan waveform (logarithmic mode)
EF	X1050	X1050, X1210	file error flag: 0, 1
E$	X1080	X1080, X1150, X1160, X1170, X1180, X1190, X1200	DOS Error #

TABLE IV-7

(CONTINUED)

VARIABLE	LINE FIRST USED	VALUES ASSIGNED AT LINES	MEANING
FA	X680	X680	Overall linear scale factor
F1	160	160, 3200, 5140	Center frequency of green channel (MHz)
F1$	4390	4390, 5290, 5310	File number
F2	160	160, 3210, 5150	Center frequency of red channel (MHz)
F2$	4400	4400, 5230, 5270	File numbers excluding center hyphen
F3	160	160, 3220, 5160	Center frequency of cyan channel (MHz)
F$	4270	4270	First trial File Number selection
H$	4120	4120, 4290	Vertical scale multiplier-linear mode (string variable)
I	various		(Integer) general index variable
IVL	160	160, 3190, 5330, X800	(Integer) Time-base duration code (0, 1, 2, 3, 4, 5)
K	various		(Integer) general index variable
K$	X1280	X1280, X1290	action selection after disk error
L	4330	4330, 4370	(Integer) variable used to test length of W$ and add leading zero if required
LD$	5370	5370	Verticle range span (logarithmic mode): 1, 2, 3, 4 decodes (string variable)
LF	250	250	(Integer) Snapshot data flag
L1	X350	X350	(Integer) first digit of horizontal starting time
L2	X360	X360	(Integer) second digit of horizontal starting time
L3	X370	X370	(Integer) third digit of horizontal starting time
L4	X380	X380	(Integer) decimal digit of horizontal starting time
MA	3770	3770, 3830	(Integer) reciprocal of the greater of A1, A2, A3
MM	160	160, 5420	(Integer) maximum size of vertical display range
N	various		(Integer) general index variable
NA	2000	3090, 3320	(Integer) numerical value of even data bytes
NB	2010	3330	(Integer) numerical value of odd data bytes

TABLE IV-7

(CONTINUED)

VARIABLE	LINE FIRST USED	VALUES ASSIGNED AT LINES	MEANING
NF	4520	4520, 5470, X960, X1050, X1080	(Integer) "new file" flag
O$	420	420, 430	"Quick" or "Full" plot choice
	1010	1010, 1020	
P	1990	1990, 3310	location of even data bytes
PA	930	3640, 3680, 3740, 3810, 5510, 5580, 5660, 5740	code for individual waveform scale modification "0" = no mod.; "1" = mod.
PB	880	4210, 5400	code for overall scale modification "0" = no mod.; "1" = mod.
PT	X250	X250	number used to compute second byte of overall scale factor representation
P$	590	590, 600	choice of "File" or "Quick Plot."
Q	2970	2970	Index used to represent byte locations in search for start address
QT	X250	X250	second byte of overall scale factor representation
QØ	2940	2940	Data start address
Q$	X490	X490	sequential characters in file # (string variable)
S(35)	150	3100	(Integer array) Display parameter codes
SC	160	160, 3180, 4030, 5410, X210	(Integer) Vertical range: 1, 2, 3, 4 decodes (logarithmic mode)
SD(4)	150	3150	(Integer array) Delay (waveform start time) digit sequence
ST	160	160	(Integer) Quick plot step size
T	160	160, 3880, 4200, 5110	Overall scale factor (linear mode)
TD	4160	4160	Vertical scale multiplying factor - linear mode
TD$	4110	4110	Overall vertical scale multiplier-first trial - linear mode (string variable)
TT	X230	X230	number used to determine first byte of overall scale factor representation
U	5250	5250	variable representing character numerical equivalent in stored file numbers

TABLE IV-7
(CONTINUED)

VARIABLE	LINE FIRST USED	VALUES ASSIGNED AT LINES	MEANING
VL	2540	2540, 2620, 2670, 2760, 2830, 2900	Horizontal time scale range in μsec
V$(Ø)	150	---	Dummy variable
W$	4260	4260, 4310, 4360	First trial File-Number selection with all but alphanumerics removed
XI	2030	2030, 3350	(Integer) amplitude representation of stored data sample (max: $2^{11} - 1$)
XJ	2060	2060, 2120, 2180, 3380, 3440, 3500	(Integer) log amplitude representation of data sample in pixels
XY	5630	5630	(Integer) log waveform amplitude shift (dB) selection
XY$	5540	5540, 5620, 5700	Log waveform amplitude shift selection (dB) (string var.)
XØ(448)	150	2050, 2070, 3370, 3390	(Integer array) sequence of green channel amplitudes (linear mode)
X1(448)	150	2110, 2130, 3430, 3450	(Integer array) sequence of red channel amplitudes (linear mode)
X2(448)	150	2170, 2190, 3490, 3510	(Integer array) sequence of cyan channel amplitudes
Y	3960	3960, 3970, 3980, 3990, 4000. 4010, 4020	Reciprocal of the numberical value of Y$
Y$	3950	3950	Selected vertical individual scale factor (linear) - string variable
Z$	4420	4420, 4430	
	4460	4460, 4490	
	4760	4760	Various cue responses (string variable)
	x950	x950	
ZZ	9050	---	Dummy argument

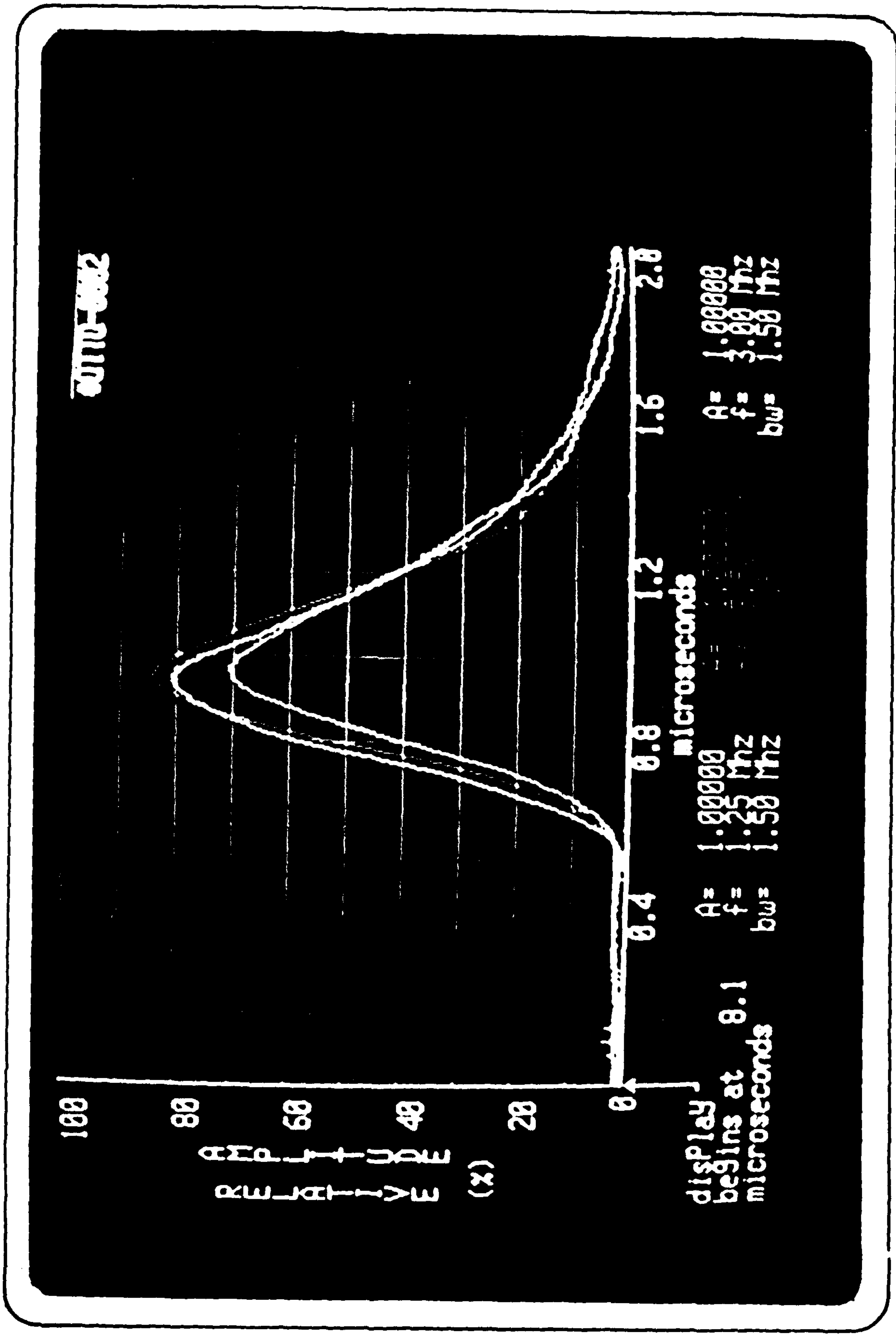


Fig. IV-4. Typical full plot display.

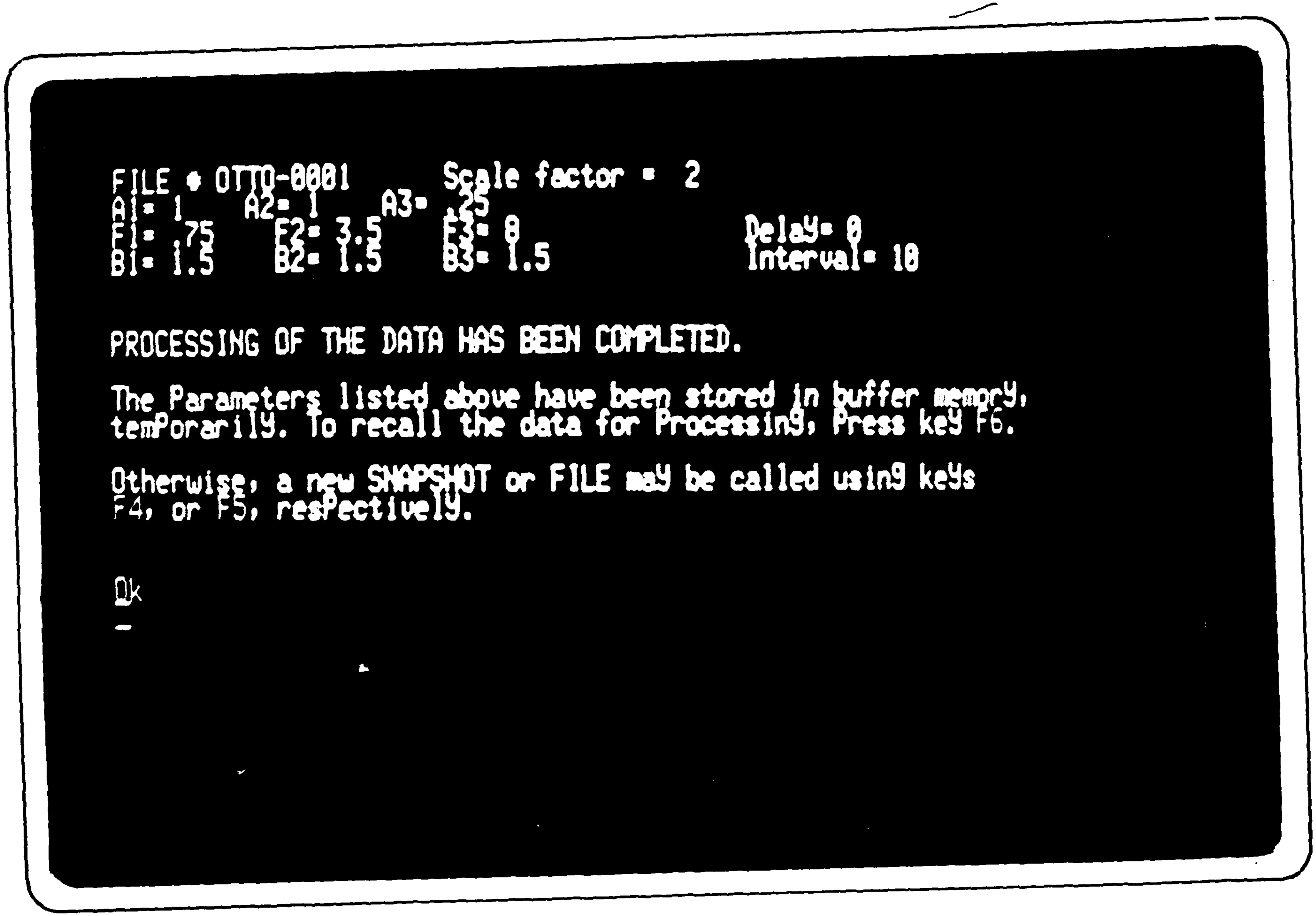


Fig. IV-5. Display after processing has been completed.

V. REFERENCE DOCUMENTATION

DEVICE	DOCUMENTATION
Appendices to this Report	
Power Supply (Filter/Digitizer)	Acopian model 51515T9 specification sheet (technical manual section III-I)
1 MHz crystal oscillator (Filter/Digitizer)	Vectron dimensional drawing - Installation, C0236 Series Osc. (technical manual section III-G)
Sample-hold amplifier (Filter/Digitizer)	Datel model SHM-5 data sheet (technical manual section III-C)
Analog-digital convertor (Filter/Digitizer)	Datel model ADC-EH12B1 data sheet (technical manual section III-C)
Logarithmic amplifier (Filter/Digitizer)	Analog Devices model 775N data sheet (technical manual section III-C)
Separate Manuals	
Ultrasonic pulser/receiver	Panametrics 5055 PR manual
Ultrasonic transducer	Panametrics specification sheet for V109 transducer, S/N 32902
Oscilloscope	Tektronix 5440 manual 5A48 manual 5B42 manual
Computer display and disk storage	Chromatics CG Series Operator's Manual CG Basic Version 3.0 Reference Manual Disk Software Manual Field Alignment Manual

APPENDIX A.

DATA SHEETS FOR FILTER/DIGITIZER SUB-MODULES

RECEIVED

FEATURES

- **200nSec. Acquisition to 0.1%**
- **350nSec. Acquisition to .01%**
- **5MHz Bandwidth**
- **.005% Linearity**
- **250 pSec. Aperture Uncertainty**

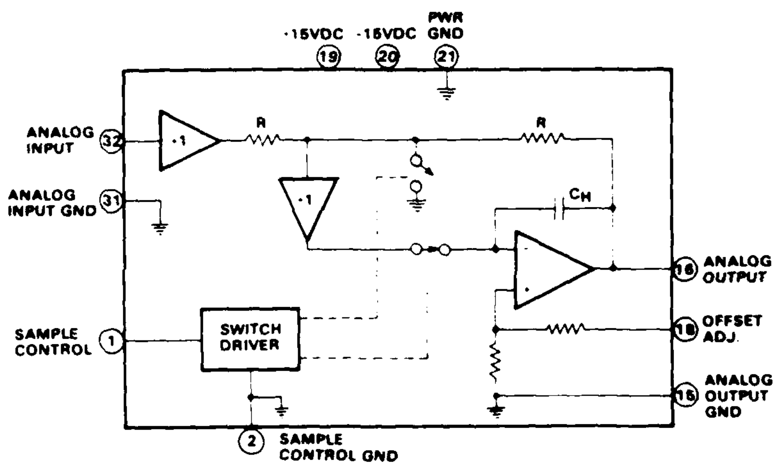
GENERAL DESCRIPTION

Model SHM-5 is a new, ultra-fast acquisition sample-hold module for use with high speed 10 and 12-bit A/D converters. When used with Datel-Intersil's model ADC-EH12B3, a 12-bit 2 µsec, A/D, the SHM-5 permits sampling and conversion at rates up to 425 kHz. The key circuit element in the SHM-5 is an ultra-fast settling hybrid operational amplifier manufactured in Datel's thin-film hybrid facility. This amplifier operates in the inverting mode as a hold amplifier. A fast FET sampling switch operates between two virtual ground points in order to keep switching errors small and independent of signal level. A second FET switch operates out-of-phase with the first one to minimize signal feedthrough errors.

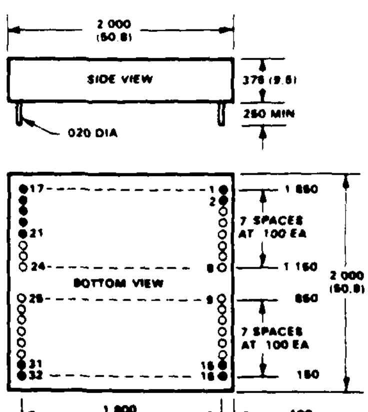
The SHM-5 is designed primarily for fast track & hold and simultaneous sampling applications with A/D converters. From the tracking mode it realizes acquisition times of 200 nsec. to 0.1% or 350 µsec. to 0.01% for a 10V change. When the input buffer amplifier must also make a 10V change, as in multiplexer applications, the total acquisition time is 1 µsec. to 0.01%.

The SHM-5 operates in the inverting mode with a gain of -1 and an input impedance of 10^8 ohms. Dynamic characteristics include a 5 MHz small signal bandwidth, and 25V/µsec. slew rate in the sampling (tracking) mode. When acquiring a new sample, however, the internal slew rate across the holding capacitor is 200V/µsec. Aperture delay time is 20 nanoseconds and aperture uncertainty time is 250 picoseconds.

This device is packaged in a 2 x 2 x 0.375 inch epoxy encapsulated module. Operating temperature range is 0°C to 70°C and power requirement is ±15VDC at 75 mA maximum. Model SHM-5 is pin compatible with Datel-Intersil's model SHM-UH3.



MECHANICAL DIMENSIONS INCHES (MM)



INPUT/OUTPUT CONNECTIONS

PIN	FUNCTION
1	SAMPLE CONTROL
2	SAMPLE CONTROL GND
15	ANALOG OUTPUT GND
16	ANALOG OUTPUT
17	NO CONNECTION
18	OFFSET ADJ.
19	+15V POWER
20	-15V POWER
21	POWER GND
31	ANALOG INPUT GND
32	ANALOG INPUT

INPUTS	
Input Voltage Range	±10V min
Input Overvoltage, no damage	±15V
Input Impedance	10^8 ohms
Input Bias Current	250nA max
Sample Control, sample mode	+2.0 to +5.5V
hold mode	0V to +0.8V
Sample Control Loading	+1mA
Offset Adjustment Range	±300mV
OUTPUT	
Output Voltage Range, min.	±10V
Output Current, S.C. protected	±40mA
Output Impedance	0.1 Ωmax
PERFORMANCE	
Gain	1.000 ±0.1%
Gain Temp. Coefficient	±15ppm/°C max
Output Offset Voltage, sample mode	±50mV max
Output Offset Voltage Drift	±30μV/°C max
Sample to Hold Offset Error	±5mV max
Tracking Nonlinearity	±.005% max
Hold Mode Droop	20μV/μsec max
Hold Mode Feedthrough, DC-500kHz	0.02%
Output Offset vs Supply	1mV/V
DYNAMIC RESPONSE	
Acquisition Time[1], 10V to 0.1%	200 nsec max
Acquisition Time[1], 10V to .01%	350 nsec max
Acquisition Time[2], 10V to 0.01%	1.0 μsec typ. 1.5 μsec max
Bandwidth, tracking, −3dB	5MHz
Slew Rate, tracking	25V/μsec
Aperture Delay Time	20nsec
Aperture Uncertainty Time	250 psec
POWER REQUIREMENT	
Power Supply Voltage	±15VDC ±0.5V
Quiescent Current	75mA max
PHYSICAL-ENVIRONMENTAL	
Operating Temp. Range	0°C to 70°C
Storage Temp. Range	−55°C to +85°C
Relative Humidity	Up to 100% non-condensing
Case Size	2.0 X 2.0 X 0.375 in 50.8 X 50.8 X 9.5 mm
Case Material	Black diallyl phthalate per MIL-M-14
Pins	.020" round gold plated .25" long min
Weight	2 oz (57g)

NOTES:

1. From tracking mode
2. From input buffer

ORDERING INFORMATION

	PRICE (1-9)
SHM-5	$218.00
Mating Socket: DILS-2 (2/module) at $5.00/pair	
Trimming Potentiometer, TP20K	$3.00

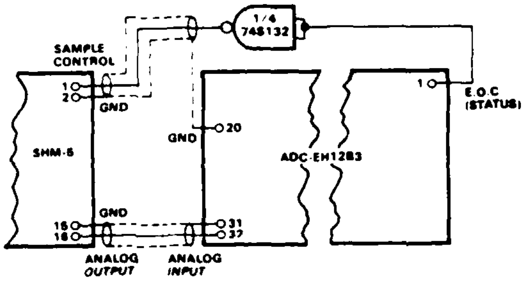
For extended temperature range operation, the following suffix is added to the model number. Consult factory for price and delivery.

- −EX −25°C to +85°C operation
- −EXX-HS −55°C to +85°C with hermetically sealed semiconductor components.

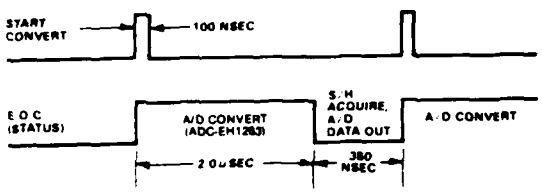
TECHNICAL NOTES

1. The SHM-5 initial gain error of ±0.1% must be adjusted out separately from the sample hold. This is most easily done by using the gain adjust of the A/D converter used with the SHM-5.
2. The maximum sample-to-hold offset error of 5mV is constant with signal level. This error can be adjusted out in the hold mode by means of the external offset adjustment shown in the diagram. It should be noted that the SHM-5 can be adjusted for zero output offset in either the sample (tracking) mode or the hold mode, but not in both at the same time.
3. The sample control input is compatible with standard TTL levels. It is recommended that this input be driven from its own active pull-up Schottky TTL circuit, such as the 74S132. This will readily supply the +1mA drive current required by the SHM-5.
4. The analog signal delay from the input of the SHM-5 to the sampling switch is approximately 32 nsec. Aperture delay is 20 nsec.
5. When the SHM-5 is switched into the hold mode, about 50 nsec. is required for the switch transient to settle. This time should be allowed for before the first A/D conversion is made.

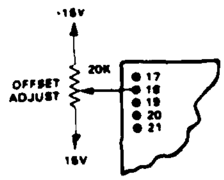
CONNECTION TO ADC-EH12B3



TIMING DIAGRAM



OFFSET ADJUSTMENT



THE SHM-5 IS COVERED BY GSA CONTRACT.

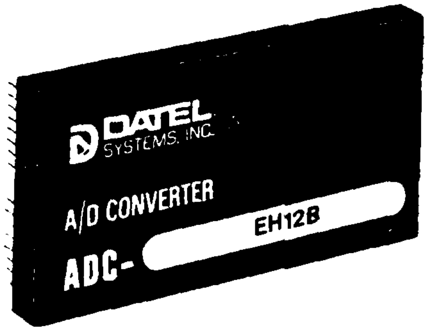
DATEL INTERSIL
11 CABOT BOULEVARD, MANSFIELD, MA 02048 / TEL (617)339-9341 / TWX 710-346-1953 / TLX 951340
Santa Ana, (714)835-2751, (L.A.) (213)933-7256 • Sunnyvale, CA (408)733-2424 • Gaithersburg, MD (301)840-9490 • Houston, (713)781-8886 • Dallas, TX (214)241-0651 OVERSEAS: DATEL (UK) LTD—TEL ANDOVER (0264)51055 • DATEL SYSTEMS SARL 602-57-11 • DATELEK SYSTEMS GmbH (089)77-60-95 • DATEL KK Tokyo 793-1031

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE

DATEL
SYSTEMS, INC.

12 BIT, 4.0 AND 8.0μSEC. ANALOG TO DIGITAL CONVERTERS

- ▶ **4.0μsec. Conversion—ADC-EH12B2**
- ▶ **8.0μsec. Conversion—ADC-EH12B1**
- ▶ **12 Bit Resolution**
- ▶ **30PPM/°C Tempco**
- ▶ **Low Profile—0.4" High**

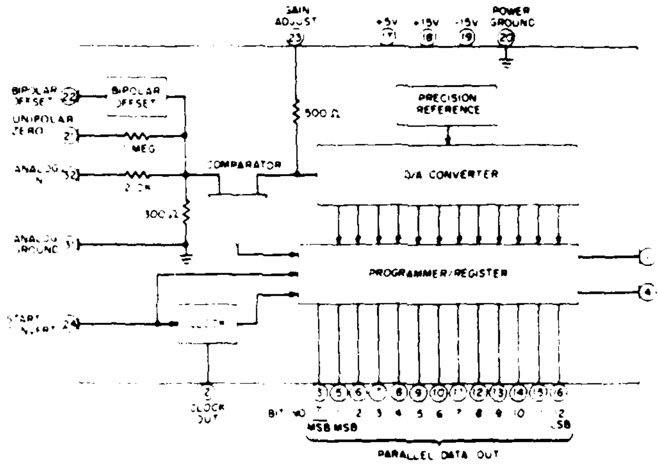


Model ADC-EH12B is a 4 microsecond, 12 bit successive approximation type A/D converter in a low profile 4 x 2 x 0.4 inch module. This high performance converter is priced at about half that of other competing models; in addition, it consumes only 2.0 watts of power, much less than competing devices. It is ideal for application in PCM systems, data acquisition systems, and other instrumentation and control systems requiring very fast data conversion rates up to 250,000 per second. The ADC-EH12B is also available in an even lower cost 8.0 μsec. version.

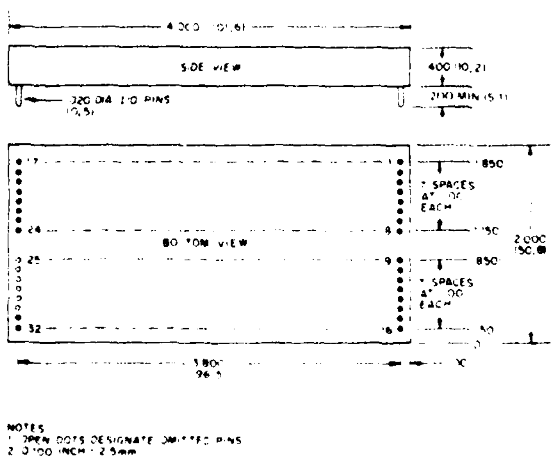
The ADC-EH12B design utilizes an MSI integrated circuit successive approximation programmer/register, 12 fast switching current sources, a low impedance R-2R resistor network, a precision voltage reference circuit, and a fast precision comparator to achieve its very fast conversion rate.

Operating features include unipolar (0 to +10V) or bipolar (±5V) operation by external pin connection. Full scale temperature coefficient is 30ppm/°C maximum and the converter is monotonic over its full operating temperature range of 0°C to 70°C. External offset and gain adjustments are provided for precise calibration of zero and full scale. Parallel output coding is straight binary for unipolar operation and offset binary or two's complement for bipolar operation. A serial output gives successive decision pulses in NRZ format with straight binary or offset binary coding. Other outputs include clock output for synchronization with serial data, $\overline{\text{MSB}}$ output for use in two's complement coding, and end of conversion (status) signal. All outputs are DTL/TTL compatible.

Power requirement is ±15VDC and +5VDC. Extended temperature range versions are also available.



MECHANICAL DIMENSIONS INCHES (MM)



NOTES
1. OPEN DOTS DESIGNATE OMITTED PINS
2. 0.100 INCH = 2.5mm

INPUT/OUTPUT CONNECTIONS

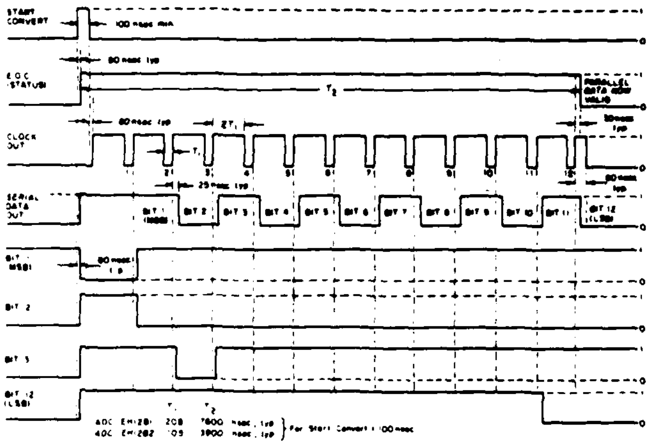
PIN	FUNCTION
1	E.O.C. (STATUS)
2	CLOCK OUT
3	BIT 1 OUT (MSB)
4	SERIAL DATA OUT
5	BIT 1 OUT (MSB)
6	BIT 2 OUT
7	BIT 3 OUT
8	BIT 4 OUT
9	BIT 5 OUT
10	BIT 6 OUT
11	BIT 7 OUT
12	BIT 8 OUT
13	BIT 9 OUT
14	BIT 10 OUT
15	BIT 11 OUT
16	BIT 12 OUT (LSB)
17	+5V POWER IN
18	+15V POWER IN
19	-15V POWER IN
20	POWER GROUND
21	UNIPOLAR ZERO
22	BIPOLAR OFFSET
23	GAIN ADJUST
24	START CONVERT IN
31	ANALOG GROUND
32	ANALOG IN

Analog Input Range	0V to +10V FS or ±5V FS
Input Impedance	2.3K ohms ±0.1%
Input Overvoltage	±20V, no damage
Start Conversion	2V min. to 5.5V max. positive pulse with duration of 100 nsec. min. Rise and fall times <500 nsec. Logic "1" resets converter. Logic "0" initiates conversion. Loading: 1 TTL load
Parallel Output Data	12 parallel lines of data held until next conversion command. V out ("0") ≤ +0.4V. V out ("1") ≥ +2.4V. Each output capable of driving up to 4 TTL loads.
Coding, Unipolar operation	Straight Binary, positive true
Bipolar operation	Offset Binary, positive true. Two's complement, positive true
Serial Output Data	NRZ successive decision pulse output generated during conversion with MSB first. Straight binary or offset binary, positive true coding. Loading 4 TTL loads
End of Conversion (EOC)	Conversion Status Signal. V out ("0") ≤ +0.4V indicates conversion completed. V out ("1") ≥ +2.4V during reset and conversion. Loading 4 TTL loads
Clock Output	Internal clock pulse train of negative going pulses from +5V to 0V gated on during conversion time. Loading 6 TTL loads
Resolution	12 Bits (1 part in 4096)
Accuracy at 25°C	±.012% of FS ±1/2 LSB.
Nonlinearity	±1/2 LSB max.
Differential Nonlinearity	±1/2 LSB max.
Differential Nonlinearity T.C.	±3ppm/°C max.
Temp. Coeff. of Gain	±30ppm/°C max.
Temp. Coeff. of Zero, unipolar	±150 µV/°C max.
Temp. Coeff. of Offset, bipolar	±15ppm of F.S./°C max.
Power Supply Rejection	.01% FS/% supply, max.
Conversion Time	8.0 µsec. max., ADC-EH12B1; 4.0 µsec. max., ADC-EH12B2
POWER REQUIREMENT	+15VDC ±0.5VDC @ 40mA max. +5VDC ±0.25VDC @ 150mA max

PHYSICAL-ENVIRONMENTAL

Operating Temp. Range	0°C to 70°C
Storage Temp. Range	-25°C to +85°C
Relative Humidity	Up to 100% non-condensing
Case Size	4 x 2 x 0.4 inches (101.6 x 50.8 x 10.2mm)
Case Material	Black Diallyl Phthalate per MIL-M-14
Pins	.020" round, gold plated, .200" long min.
Weight	4 oz. max. (114 g.)

TIMING DIAGRAM FOR ADC-EH12B Output: 101010101010



OUTPUT CODING

UNIPOLAR (0V TO +10V)

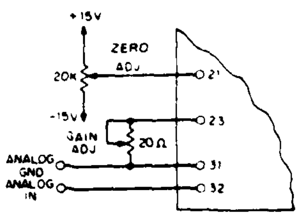
SCALE	INPUT VOLTAGE	STRAIGHT BINARY
+FS - 1 LSB	+9.9976V	1111 1111 1111
+7/8 FS	+8.7500V	1110 0000 0000
+3/4 FS	+7.5000V	1100 0000 0000
+1/2 FS	+5.0000V	1000 0000 0000
+1/4	+2.5000V	0100 0000 0000
+1 LSB	+0.0024V	0000 0000 0001
0	0.0000V	0000 0000 0000

BIPOLAR (−5V TO +5V)

SCALE	INPUT VOLTAGE	OFFSET BINARY	TWO'S COMPLEMENT*
+FS - 1 LSB	+4.9976V	1111 1111 1111	0111 1111 1111
+3/4 FS	+3.7500V	1110 0000 0000	0110 0000 0000
+1/2 FS	+2.5000V	1100 0000 0000	0100 0000 0000
0	0.0000V	1000 0000 0000	0000 0000 0000
-1/2 FS	-2.5000V	0100 0000 0000	1100 0000 0000
-3/4 FS	-3.7500V	0010 0000 0000	1010 0000 0000
-FS + 1 LSB	-4.9976V	0000 0000 0001	1000 0000 0001
-FS	-5.0000V	0000 0000 0000	1000 0000 0000

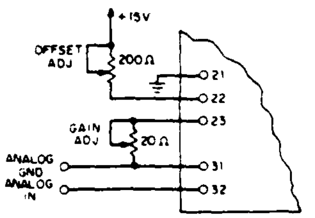
*Using MSB output for Bit 1

GAIN & OFFSET ADJUSTMENTS



UNIPOLAR OPERATION

1. Apply START CONVERT pulses to pin 24 (see specifications and timing diagram)
2. Apply a precision reference voltage source to ANALOG IN (pin 32) and ANALOG GROUND (pin 31). Adjust the output of the voltage reference to *Zero* +1/2 LSB (+1.2mV). Adjust the zero trimming potentiometer so that the output code flickers equally between 0000 0000 0000 and 0000 0000 0001
3. Adjust the output of the voltage reference to +FS - 1 1/2 LSB (+9.9963V). Adjust the GAIN trimming potentiometer so that the output code flickers equally between 1111 1111 1110 and 1111 1111 1111



BIPOLAR OPERATION

1. Apply START CONVERT pulses to pin 24 (see specifications and timing diagram)
2. Apply a precision reference voltage source to ANALOG IN (pin 32) and ANALOG GROUND (pin 31). Adjust the output of the voltage reference to -FS +1/2 LSB (-4.9988V). Adjust the offset trimming potentiometer so that the output code flickers equally between 0000 0000 0000 and 0000 0000 0001
3. Adjust the output of the voltage reference to +FS - 1 1/2 LSB (+4.9854V). Adjust the GAIN trimming potentiometer so that the output code flickers equally between 1111 1111 1110 and 1111 1111 1111

ORDERING INFORMATION

ADC-EH12B—

CONVERSION TIME
1= 8.0 µsec.
2= 4.0 µsec.

PRICES (1-9)

ADC-EH12B1	$189.00
ADC-EH12B2	$229.00

MATING SOCKETS:
DILS-2 (2/MODULE) $5.00/PAIR
TRIMMING POTENTIOMETERS:
TP20, TP200, TP20K $3.00 EACH

For extended temperature range operation, the following suffixes are added to the model number. Consult factory for pricing.

−EX	-25°C to +85°C operation
−EXX−HS	-55°C to +85°C operation with hermetically sealed semiconductor components

THE ADC-EH12B CONVERTERS ARE COVERED BY GSA CONTRACT.

DATEL SYSTEMS, INC. 11 CABOT BOULEVARD, MANSFIELD, MA 02048 / TEL. (617)828-8000 / (617)339-9341 / TWX 710-346-1953 / TLX 951340
Santa Ana, (714)835-2751 • L.A. (213)933-7256 • Sunnyvale, CA (408)733-2424 • Gaithersburg, MD (301)840-9490 • Houston, (713)932-1130 • Dallas, TX (214)241-0651
OVERSEAS: DATEL (UK) LTD—TEL: ANDOVER (0264)51055 • DATEL SYSTEMS SARL 620-06-74 • DATELEK SYSTEMS GmbH (089)77-60-95

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE

2/79 BULLETIN EHCBL 05902

ANALOG DEVICES

6-Decade, High Accuracy Log, Antilog Amplifiers

FEATURES

Complete Log/Antilog Amplifier.
- **External Components Not Required;**
- **Internal Reference; Temperature Compensated**

6 Decades Current Operation – 1nA to 1mA
- **1/2% max Error – 10nA to 100μA**
- **1% max Error – 1nA to 1mA**

4 Decades Voltage Operation – 1mV to 10V
- **1/2% max Error – 1mV to 1V**
- **1% max Error – 1mV to 10V**

APPLICATIONS

Log Current or Voltage
Antilog Voltage
Data Compression or Expansion
Absorbence Measurements
Computing Powers and Log Ratios

GENERAL DESCRIPTION

Model 755 is a complete dc logarithmic amplifier consisting of an accurate temperature compensated antilog element, and a low bias current FET amplifier. In addition to offering 120dB of current logging (1nA to 1mA) and 80dB of voltage logging (1mV to 10V), the 755 features exceptionally low bias currents of 10pA and 15μV/°C voltage drift to satisfy most wide range applications. Conformance to ideal log operation is held to ±1% over its total 120dB current range (1nA to 1mA), with ±0.5% conformity guaranteed over an 80dB range (10nA to 100μA). Two models are available, model 755N and model 755P. The N version computes the log of positive input signals and the P version computes the log of negative input signals.

Advanced design techniques and improved component selection are used to obtain exceptionally good performance. For example, the use of monolithic devices greatly reduces the influence of temperature variations. Offering both log and antilog operation, model 755's price and performance are especially attractive as an alternative to in-house designs of OEM applications. This log design also improves significantly over competitive designs in price, performance, and package size.

MAJOR IMPROVEMENTS IN I_{os}

For most low level applications, the input bias current I_{os}, is especially critical, since it is the major source of error when processing low level currents. At 1nA of input current there is an error contribution of 1% for every 10pA of I_{os}. Recognizing the importance of this parameter, bias current of model 755 is maintained below 10pA.

APPLICATIONS

When connected in the current or voltage logging configuration, as shown in Figure 1, the model 755 may be used in several key applications. A plot of input current versus output voltage is also presented to illustrate the log amplifier's transfer characteristics.

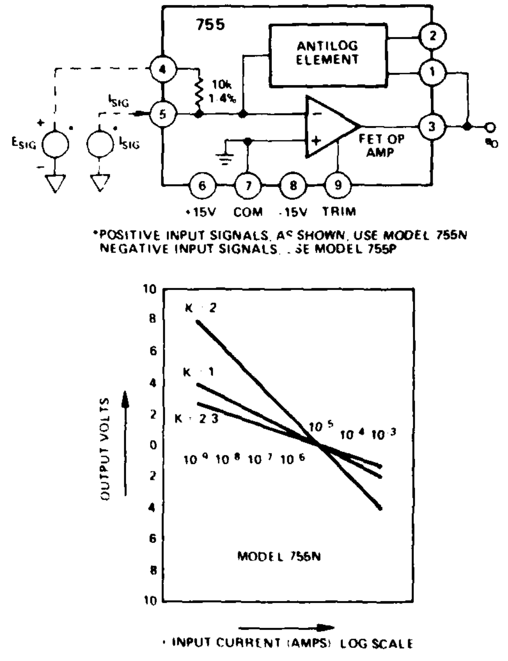


Figure 1 Functional Block Diagram and Transfer Function

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use; nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

P.O. Box 280; Norwood, Massachusetts 02062 U.S.A.
Telex: 924491 Cables: ANALOG NORWOODMASS

SPECIFICATIONS (typical @ +25°C and ±15V dc unless otherwise noted)

MODEL	755N/P
TRANSFER FUNCTIONS	
Current Mode	$e_O = -K\log_{10}\frac{I_{SIG}}{I_{REF}}$
Voltage Mode	$e_O = -K\log_{10}\frac{E_{SIG}}{E_{REF}}$
Antilog Mode	$e_O = E_{REF}\,10^{\left(\frac{E_{SIG}}{K}\right)}$
TRANSFER FUNCTION PARAMETERS	
Scale Factor (K) Selections[1, 2]	2, 1, 2/3 Volt/Decade
Error @ +25°C	±1% max
vs. Temperature (0 to +70°C)	±0.04%/°C max
Reference Voltage (E_{REF})[2]	0.1V
Error @ +25°C	±3% max
vs. Temperature (0 to +70°C)	±0.1%/°C max
Reference Current (I_{REF})[2]	10µA
Error @ +25°C	±3% max
vs. Temperature (0 to +70°C)	±0.1%/°C max

LOG CONFORMITY ERROR

I_{SIG} Range	E_{SIG} Range	R.T.I.	R.T.O. (K = 1)
1nA to 10nA		±1% max	±4.3mV max
10nA to 100µA	1mV to 1V	±0.5% max	±2.17mV max
100µA to 1mA	1V to 10V	±1% max	±4.3mV max
1nA to 1mA		±1% max	±4.3mV max

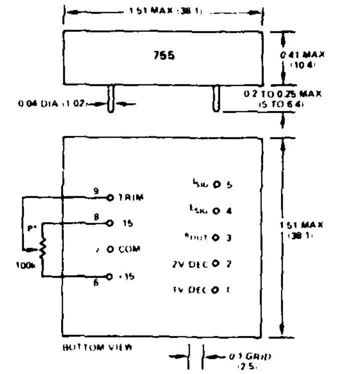
INPUT SPECIFICATIONS	
Current Signal Range	
Model 755N	+1nA to +1mA min
Model 755P	-1nA to -1mA min
Max Safe Input Current	±10mA max
Bias Current @ +25°C	(0, +) 10pA max
vs. Temperature (0 to +70°C)	x2/+10°C
Voltage Signal Range (Log Mode)	
Model 755N	+1mV to +10V min
Model 755P	-1mV to -10V min
Voltage Signal Range, Antilog Mode Model 755N, 755P	$-2 \le \frac{E_{SIG}}{K} \le 2$
Offset Voltage @ +25°C (Adjustable to 0)	±400µV max
vs. Temperature (0 to +70°C)	±15µV/°C max
vs. Supply Voltage	±15µV/%
FREQUENCY RESPONSE, Sinewave	
Small Signal Bandwidth, -3dB	
I_{SIG} = 1nA	80Hz
I_{SIG} = 1µA	10kHz
I_{SIG} = 10µA	40kHz
I_{SIG} = 1mA	100kHz
RISE TIME	
Increasing Input Current	
10nA to 100nA	100µs
100nA to 1µA	7µs
1µA to 1mA	4µs
Decreasing Input Current	
1mA to 1µA	7µs
1µA to 100nA	30µs
100nA to 10nA	400µs
INPUT NOISE	
Voltage, 10Hz to 10kHz	2µV rms
Current, 10Hz to 10kHz	2pA rms
OUTPUT SPECIFICATIONS[3]	
Rated Output	
Voltage	±10V min
Current	
Log Mode	±5mA
Antilog Mode	±4mA
Resistance	0.5Ω
POWER SUPPLY[4]	
Rated Performance	±15V dc
Operating	±(12 to 18)V dc
Current, Quiescent	±7mA
TEMPERATURE RANGE	
Rated Performance	0 to +70°C
Operating	-25°C to +85°C
Storage	-55°C to +125°C
CASE SIZE	1.5" x 1.5" x 0.4"

[1] Use terminal 1 for K = 1V/decade, terminal 2 for K = 2V/decade, terminals 1 or 2 (shorted together) for K = 2/3V/decade.
[2] Specification is + for model 755N, - for model 755P.
[3] No damage due to any pin being shorted to ground.
[4] Recommended power supply, model 904, ±15V @ ±50mA output.

Specifications subject to change without notice.

OUTLINE DIMENSIONS

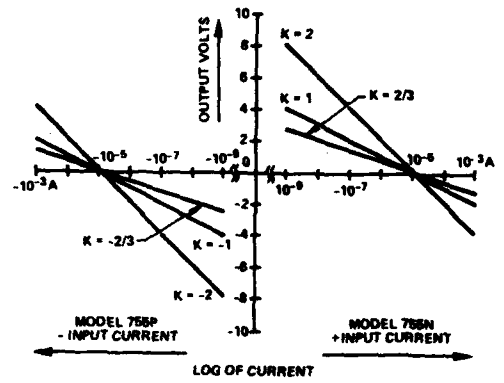
Dimensions shown in inches and (mm).



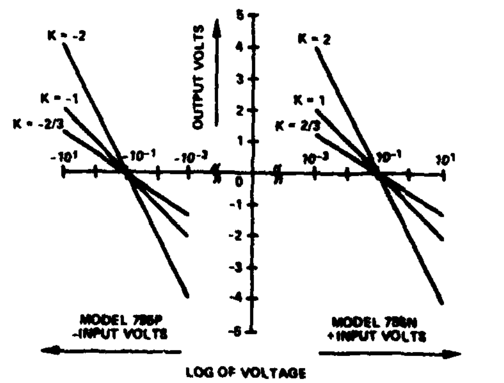
*Optional 100kΩ external trim pot – ADI PN79PR100k. Input offset voltage may be adjusted to zero with trim pot connected as shown. With trim terminal 9 left open, input offset voltage will be ±0.4mV maximum.

MATING SOCKET AC1016

TRANSFER CURVES



Plot of Output Voltage vs Input Current for Model 755 Connected in the Log Mode



Plot of Output Voltage vs Input Voltage for Model 755 Connected in the Log Mode

PRINCIPLE OF OPERATION

Log operation is obtained by placing the antilog element in the feedback loop of the op amp as shown in Figure 1. At the summing junction, terminal 5, the input signal current to be processed is summed with the output current of the antilog element. To attain a balance of these two currents, the op amp provides the required output voltage to the antilog feedback element. Under these conditions the ideal transfer equation (K = 1) is:

$$e_{OUT} = 1V \log_{10} I_{SIG}/I_{REF}$$

The log is a mathematical operator which is defined only for numbers, which are dimensionless quantities. Since an input current would have the dimensions of amperes it must be referenced to another current, I_{REF}, the ratio being dimensionless. For this purpose a temperature compensated reference of 10μA is generated internally.

The scale factor, K, is a multiplying constant. For a change in input current of one decade (decade = ratio of 10:1), the output changes by K volts. K may be selected as 1V or 2V by connecting the output to pin 1 or 2, respectively. If the output is connected to both pins 1 and 2, K will be 2/3V.

A graph of the ideal transfer function for model 755N is presented in Figure 2, for one decade of operation. Although specific values of i_{in} and e_{out} are presented for n = 1, other values may be plotted by varying n.

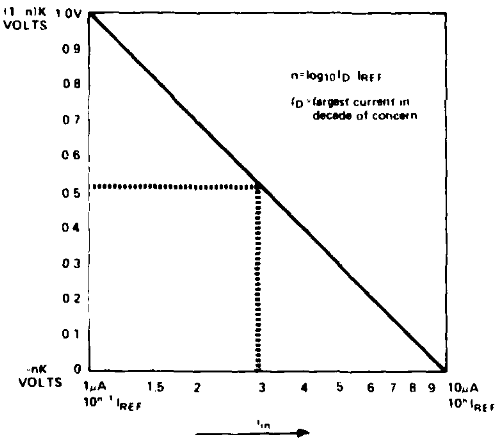


Figure 2. Input vs. Output for Any One Decade of Operation

REFERRING ERRORS TO INPUT

A unique property of log amplifiers is that a dc error of any given amount at the output corresponds to a constant percent of the input, regardless of input level. To illustrate this, consider the output effects due to changing the input by 1%.

The output would be:

$$e_{out} = 1V \log_{10} (I_{SIG}/I_{REF})(1.01) \text{ which is equivalent to}$$

$$e_{out} = \underbrace{1V \log_{10} I_{SIG}/I_{REF}}_{\text{Initial Value}} \quad \underbrace{\pm 1V \log_{10} 1.01}_{\text{Change}}$$

The change in output, due to a 1% input change is a constant value of ±4.3mV. Conversely, a dc error at the output of ±4.3mV is equivalent to a change at the input of 1%. An abbreviated table is presented below for converting between errors referred to output (R.T.O.), and errors referred to input (R.T.I.).

TABLE 1

Error. R.T.I. (N)	Error R.T.O. K = 1	K = 2	K = 2/3
0.1%	0.43mV	0.86mV	0.28mV
0.5	2.17	4.34	1.45
1.0	4.32	8.64	2.88
3.0	12.84	25.68	8.56
4.0	17.03	34.06	11.35
5.0	21.19	42.38	14.13
10.0	41.39	82.78	27.59

Table 1. Converting Output Error in mV to Input Error in %

NOTE:
Data may be interpolated with reasonable accuracy, for small errors by adding various values of N and their corresponding R.T.O. terms. That is, for N = 2.5% and K = 1, combine 2% and 0.5% terms to obtain 10.77mV.

SOURCES OF ERROR

When applying the model 755, a firm understanding of error sources associated with log amplifiers is beneficial for achieving maximum performance. The definitions, limiations and compensation techniques for errors specified on log amplifiers will be discussed here.

Log Conformity Error – Log conformity in logarithmic devices is a *specification similar to linearity in linear devices.* Log conformity error is the difference between the value of the transfer equation and the actual value which occurs at the output of the log module, after scale factor, reference and offset errors are eliminated or taken into account. For model 755, the best linearity performance is obtained in the middle 4 decades (10nA to 100μA). For this range, log conformity error is ±0.5% R.T.I. or 2.17mV R.T.O. To obtain optimum performance, the input data should be scaled to this range.

Offset Voltage (E_{os}) – The offset voltage, E_{os}, of model 755 is the offset voltage of the internal FET amplifier. This voltage appears as a small dc offset voltage in series with the input terminals. For current logging applications, its error contribution is negligible. However, for log voltage applications, best performance is obtained by an offset trim adjustment.

Offset Current (I_{os}) – The offset current, I_{os}, of model 755 is the bias current of the internal FET amplifier. This parameter can be a significant source of error when processing signals in the nanoamp region. For this reason, I_{os}, for model 755, is held within a conservative 10pA max.

Reference Current (I_{REF}) – I_{REF} is the internally generated current source to which all input currents are compared. I_{REF} tolerance errors appear as a dc offset at the output. The specified value of I_{REF} is ±3%, referred to the input, and, from Table 1, corresponds to a dc offset of ±12.84mV, for K = 1. This offset is independent of input signal and may be removed by injecting a current into terminal 1 or 2.

Reference Voltage (E_{REF}) – E_{REF} is the effective internally generated voltage to which all input voltages are compared. It is related to I_{REF} through the equation:
$E_{REF} = I_{REF} \times R_{in}$, where R_{in} is an internal 10kΩ, precision resistor. Virtually all tolerance in E_{REF} is due to I_{REF}. Consequently, variations in I_{REF} cause a shift in E_{REF}.

Scale Factor (K) – Scale factor is the voltage change at the output for a decade (i.e., 10:1) change at the input, when connected in the log mode. Error in scale factor is equivalent to a change in gain, or slope, and is specified in per cent of the nominal value. An external adjustment may be performed if fine trimming is desired for improved accuracy.

EXTERNAL ADJUSTMENTS FOR LOG OPERATION (OPTIONAL)

Trimming E_{os} The amplifier's offset voltage, E_{os}, may be trimmed for improved accuracy with the model 755 connected in its log circuit. To accomplish this, a 100kΩ, 10 turn pot is connected as shown in Figure 3, and the input terminal, pin 4, is connected to ground. Under these conditions the output voltage is:

$$e_{out} = -K \log_{10} E_{os}/E_{REF}$$

To obtain an offset voltage of 100μV or less, for K = 1, the trim pot should be adjusted until the output voltage is between +3 and +4 volts for model 755N, and -3 to -4V for model 755P.

For other values of K, the trim pot should be adjusted for an output of $e_{out} = 3 \times K$ to $4 \times K$ where K is the scale factor.

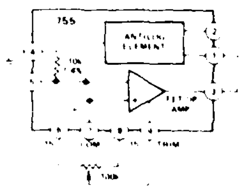


Figure 3. Trimming E_{OS} in Log Mode

REFERENCE CURRENT OR REFERENCE VOLTAGE

The reference current or voltage of model 755 may be shifted by injecting a constant current into the unused scale factor terminal (pin 1 or pin 2). Each 66μA of current injected will shift the reference one decade, in accordance with the expression: $I_I = 66\mu A \log 10\mu A/I_{REF}$, where I_I = current to be injected and I_{REF} = the desired reference current.

By changing I_{REF}, there is a corresponding change in E_{REF} since, $E_{REF} = I_{REF} \times R_{in}$. An alternate method for rescaling E_{REF} is to connect an external R_{in}, at the I_{in} terminal (pin 5) to supplant the 10kΩ supplied internally (leaving it unconnected). The expression for E_{REF} is then, $E_{REF} = R_{in} I_{REF}$. Care must be taken to choose R_{in} such that $(e_{in}\ max)/R_{in} \leq 1mA$.

Scale Factor (K) Adjustment Scale factor may be increased from its nominal value by inserting a series resistor between the output terminal, pin 3, and either terminal 1 or 2. The table below should be consulted when making these scale factor changes.

TABLE 2

Range of K	Connect Series R to Pin	Value of R	Note
2/3V to 1.01V	1	15kΩ x (K - 2/3)	use pins 1, 2
1.01V to 2.02V	1	15kΩ x (K - 1)	use pin 1
>2.02V	2	15kΩ x (K - 2)	use pin 2

Table 2. Resistor Selection Chart for Shifting Scale Factor

ANTILOG OPERATION

The model 755 may be used to develop the antilog of the input voltage when connected as shown in Figure 4. The antilog transfer function (an exponential), is:
$e_{out} = E_{REF}\ 10^{-e_{in}/K}$

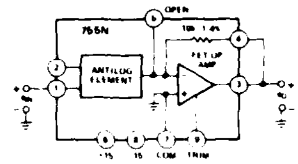


Figure 4. Functional Block Diagram

Principle of Operation – The antilog element converts the voltage input, appearing at terminal 1 or 2, to a current which is proportional to the antilog of the applied voltage. The current-to-voltage conversion is then completed by the feedback resistor in a closed-loop op amp circuit.

A more complete expression for the antilog function is:

$$e_{out} = E_{REF}\ 10^{-e_{in}/K} + E_{OS}$$

The terms K, E_{OS}, and E_{REF} are those described previously in the LOG section.

Offset Voltage (E_{os}) Adjustment – Although offset voltage of the antilog circuit may be balanced by connecting it in the log mode, and using the technique described previously, it may be more advantageous to use the circuit of Figure 5. In this configuration, offset voltage is equal to $e_{out}/100$. Adjust for the desired null, using the 100k trim pot. After adjusting, turn power off, remove the external 100Ω resistor, and the jumper from pin 1 to +15V. For 755P, use the same procedure but connect pin 1 to -15V.

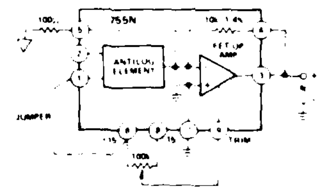


Figure 5. Trimming E_{OS} in Antilog Mode

Reference Voltage (E_{REF}) Adjustment – In antilog operation, the voltage reference appears as a multiplying constant. E_{REF} adjustment may be accomplished by connecting a resistor, R, from pin 5 to pin 3, in place of the internal 10kΩ. The value of R is determined by:

$$R = E_{REF}\ \text{desired}/10^{-5}\ A$$

Scale Factor (K) Adjustment The scale factor may be adjusted for all values of K greater than 2/3V by the techniques described in the log section. If a value of K, less than 2/3V is desired for a given application, an external op amp would be required as shown in Figure 6. The ratio of the two resistors is approximately:

$$R1/R_G = (1/K - 1) \text{ where } K = \text{desired scale factor}$$

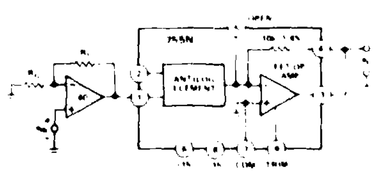
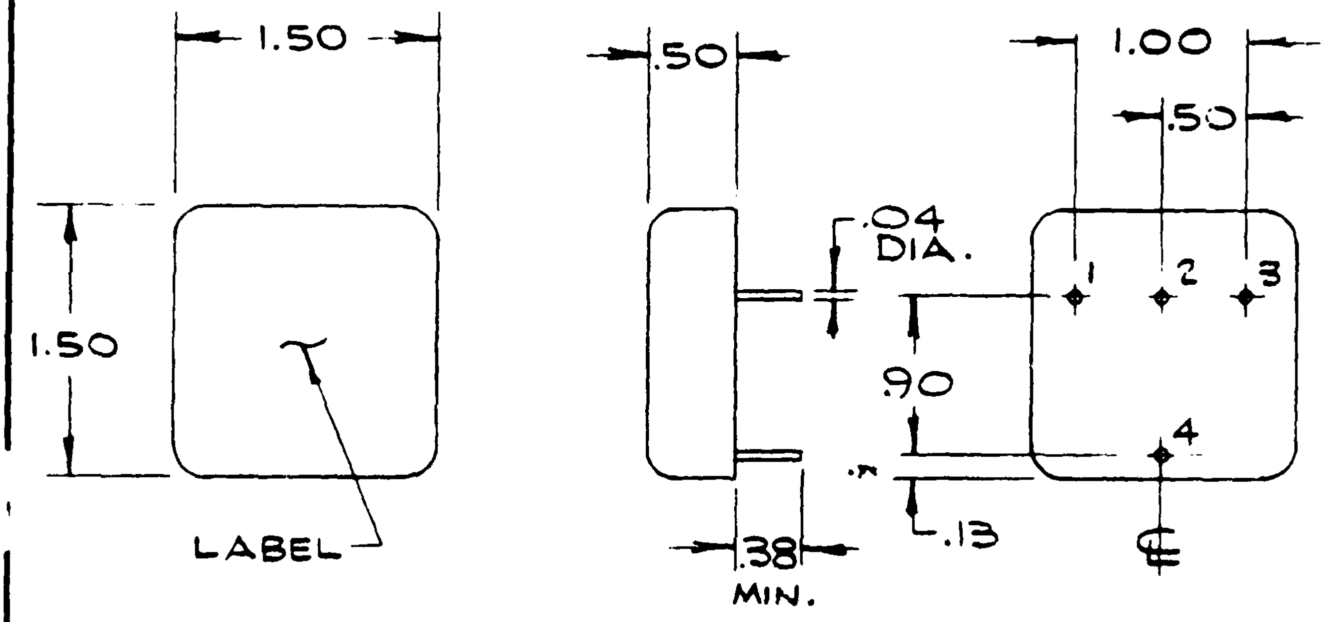


Figure 6. Method for Adjusting K<2/3V

REVISIONS				
ZONE	LTR	DESCRIPTION	DATE	APPROVED



NOTE
1. PIN NUMBERS ARE FOR REF. ONLY THEY DO NOT APPEAR ON UNIT.

TERM	FUNCTION
1	B- & RF RETURN
2	CASE
3	B+
4	RF OUTPUT

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES ON
FRAC ± 1/64 | 2 PLACE DEC DIM ± .020 | 3 PLACE DEC DIM ± .005 | ANGLES ± 30'

MATERIAL

FINISH

DO NOT SCALE DRAWING

DRAWING STARTED | DATE
DRAWN S. Marino 5/11/71
CHECKED
ENGR

VECTRON
VECTRON LABORATORIES, INC.
NORWALK, CONN

INSTALLATION
CO236 SERIES OSC

SIZE A | CODE IDENT NO. 27802 | 236-00-006

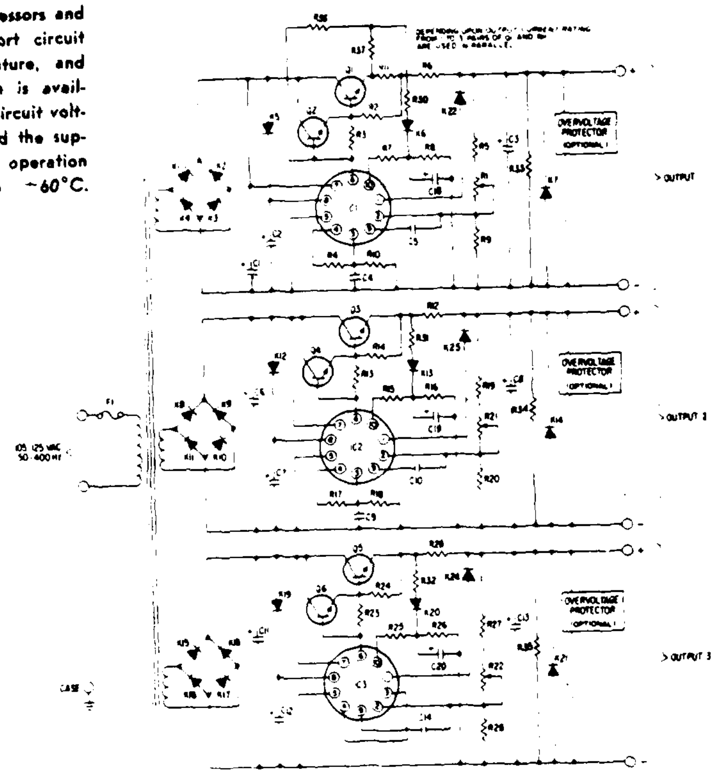
SCALE FULL | SHEET

Acopian

MODEL 51515T9

TRIPLE OUTPUT POWER SUPPLY

A convenient source of the voltages required for powering microprocessors and related circuits. Electronic short circuit protection is a standard feature, and built-in overvoltage protection is available as an option. Integrated circuit voltage regulators are utilized, and the supply is conservatively rated for operation at ambient temperatures to ~60°C. without derating.



R1, R21, R22-	2.5K pot
R2, R14, R24 -	150 ohm ½W 10%
R3, R13, R23, R36 -	560 ohm ½W 10%
R4 -	3K ½W 5% metal film
R5 -	3.6K ½W 5% metal film
R6 -	short
R7, R15, R25 -	3.3K ½W 10%
R8, R16, R26 -	47K ½W 10%
R9 -	5.6K ½W 5% metal film
R10 -	2K ½W 5% metal film
R11 -	.15 ohm 5W 10%
R12, R29 -	1 ohm 5W 10%
R17 -	short
R18 -	omit
R19, R20 -	6.8K ½W 5% metal film
R27, R28 -	6.8K ½W 5% metal film
R30, R31, R32 -	22 ohm ½W 10%
R33, R34, R35 -	1.5K ½W 10%
R37 -	1K ½W 10%

K1 to K4 -	100 piv 25 amp bridge
K5 to K7 -	100 piv 1A 1N4CC2
K8 to K11 -	100 piv 1A 1N4002
K12 to K14 -	100 piv 1A 1N4002
K15 to K18 -	100 piv 1A 1N4C02
K19 to K24 -	100 piv 1A 1N4C02
IC1, IC2 -	µA723
C1 -	25C00 ufd/25V
C2 -	47 ufd/63V
C3, C8, C13 -	100 ufd/40V
C4, C15, C16, C17	.47 ufd/35V
C5, C10, C14 -	.01 ufd/250V
C6, C11 -	2000 ufd/35V
C7, C12 -	22 ufd/63V
C9 -	omit
C18, C19, C20 -	2.2 ufd/63V
Q1 -	(2) 2N3055
Q2 -	MJE3055 Motorola
Q3, Q5 -	2N3055
Q4, Q6 -	2N3053
T1 -	51515T9
F1 -	2A slo-blo

INSTALLATION AND OPERATION

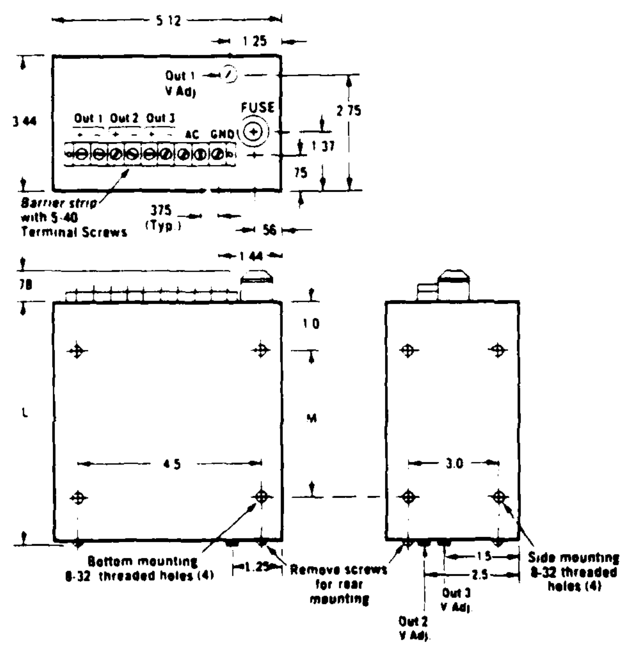
Threaded holes on the bottom and one side surface may be used for mounting, or the supply may be rear mounted by means of the threaded holes used to attach the rear cover plate.

> NOTE: Even a relatively small amount of air flowing around and through a power supply will significantly reduce the rise in its temperature resulting from operation, and therefore the temperature of the critical components within it, improving both reliability and stability. Avoid blocking air flow through vented surfaces.
>
> If the perforated bottom of a supply is mounted to a solid surface use spacers at least 3/16" thick between it and the surface to which it is fastened, to permit convection air flow, or punch ventilation holes in the mounting surface.
>
> Allow free air to circulate around heat sinks. Space surrounding objects at least 1" away.

Make all connections before applying AC input power to the supply.

If the AC input power contains large voltage spikes ("noise") induced by the switching of high currents, inductive loads, electro-mechanical components, etc., the input power leads to the supply should include some means of transient suppression. Otherwise, a portion of the noise may be coupled through the supply into the load. Also, the supply could be damaged. The means of suppression that is easiest to install is a one or two mfd. capacitor across the AC input terminals of the supply. In extremely severe cases, the use of RF chokes in series with each side of the line may also be required.

Answers to your questions regarding any Acopian product, as well as assistance with your specific needs, can be furnished by an Acopian Applications Engineer. Call Area Code 215, 258-5441 or write to Acopian Corp., Easton, Pa. 18042.



Models with Output 1 rated at 2 amps:
L = 5.09
M = 3.00

Models with Output 1 rated at 3 amps:
L = 6.59
M = 4.00

Models with Output 1 rated at 6 amps:
L = 9.25
M = 6.00

WARRANTY

Acopian products are covered by a fi five-year warranty which includes the cost of both parts and labor. The warranty is honored at the factory Service Department in Easton, Pa. If the failure has been caused by misuse, operation in excess of specifications, or modifications by the customer, repairs will be billed at cost; in such cases, a cost estimate will be submitted before work is started.

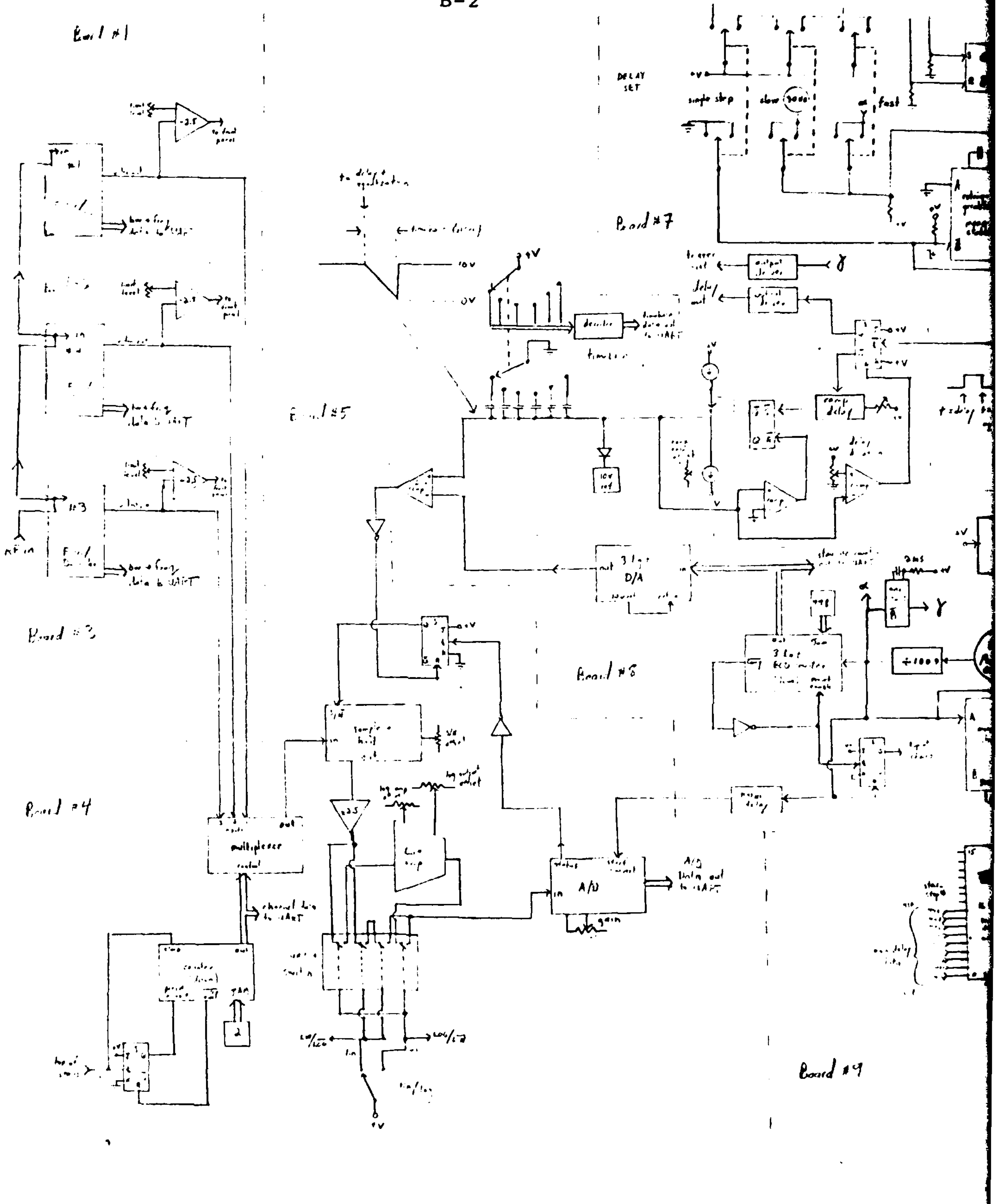
Power supplies returned to the factory for repair must have shipping charges prepaid and must be accompanied by documentation explaining the reason for the return. Written authorization must be obtained from Acopian prior to returning any goods for warranty examination.

Acopian reserves the right to make changes or improvements in its products without incurring any obligation with respect to products previously manufactured.

This warranty is in lieu of all other obligations, expressed or implied.

APPENDIX B.

FILTER/DIGITIZER SCHEMATICS AND DIAGRAMS


DELAY SET
single step
slow
fast
Board #7
Board #5
Board #3
Board #4
Board #8
Board #9
10V
0V
multiplexer
sample & hold
3 bit D/A
3 bit BCD counter
A/D
÷1000
x2.5

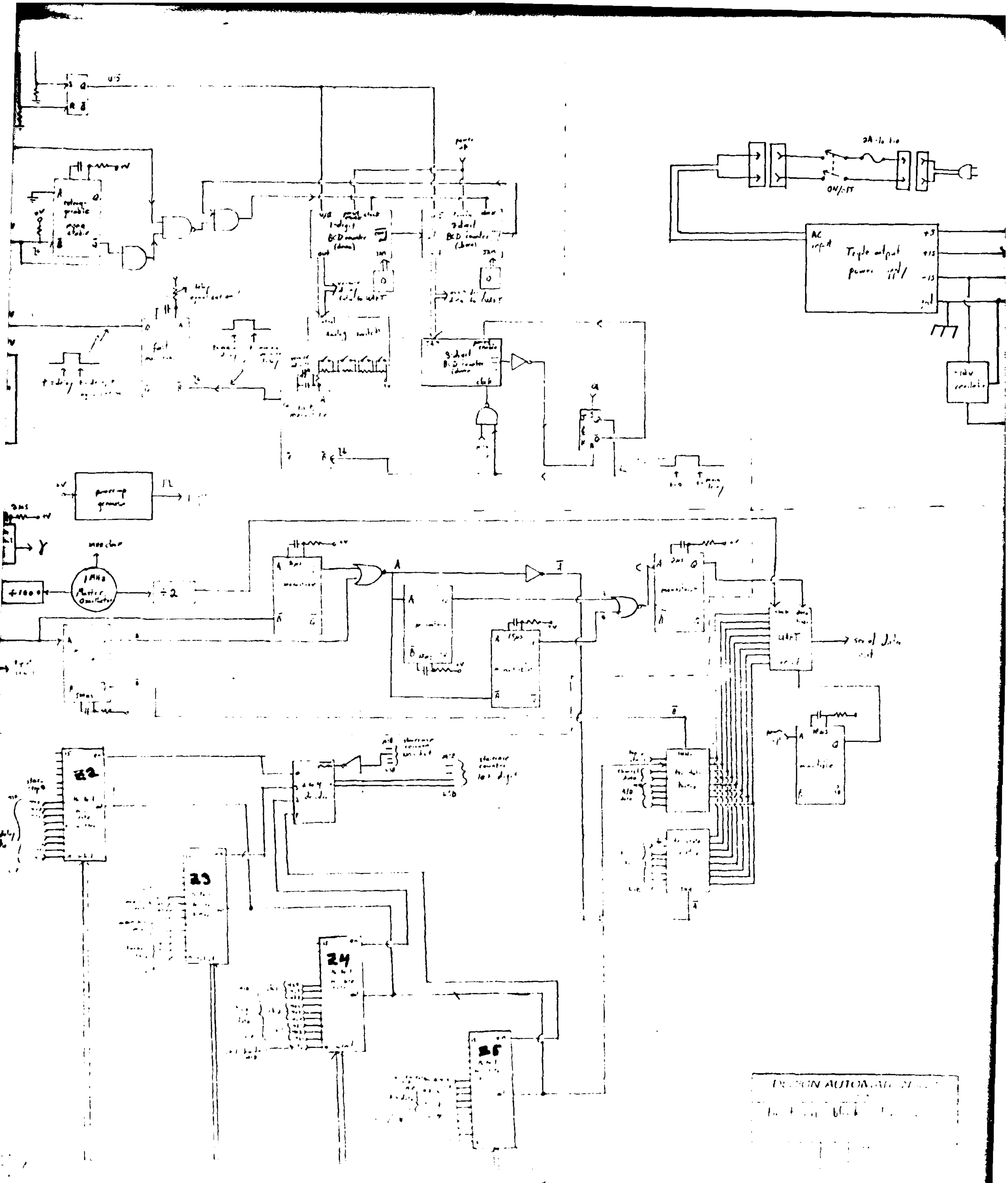

Triple output power supply
ON/OFF
1 MHz Master Oscillator
÷2
+100
power-up generator
UART
monostable
Z2
Z3
Z4
Z5

Fig. B-1

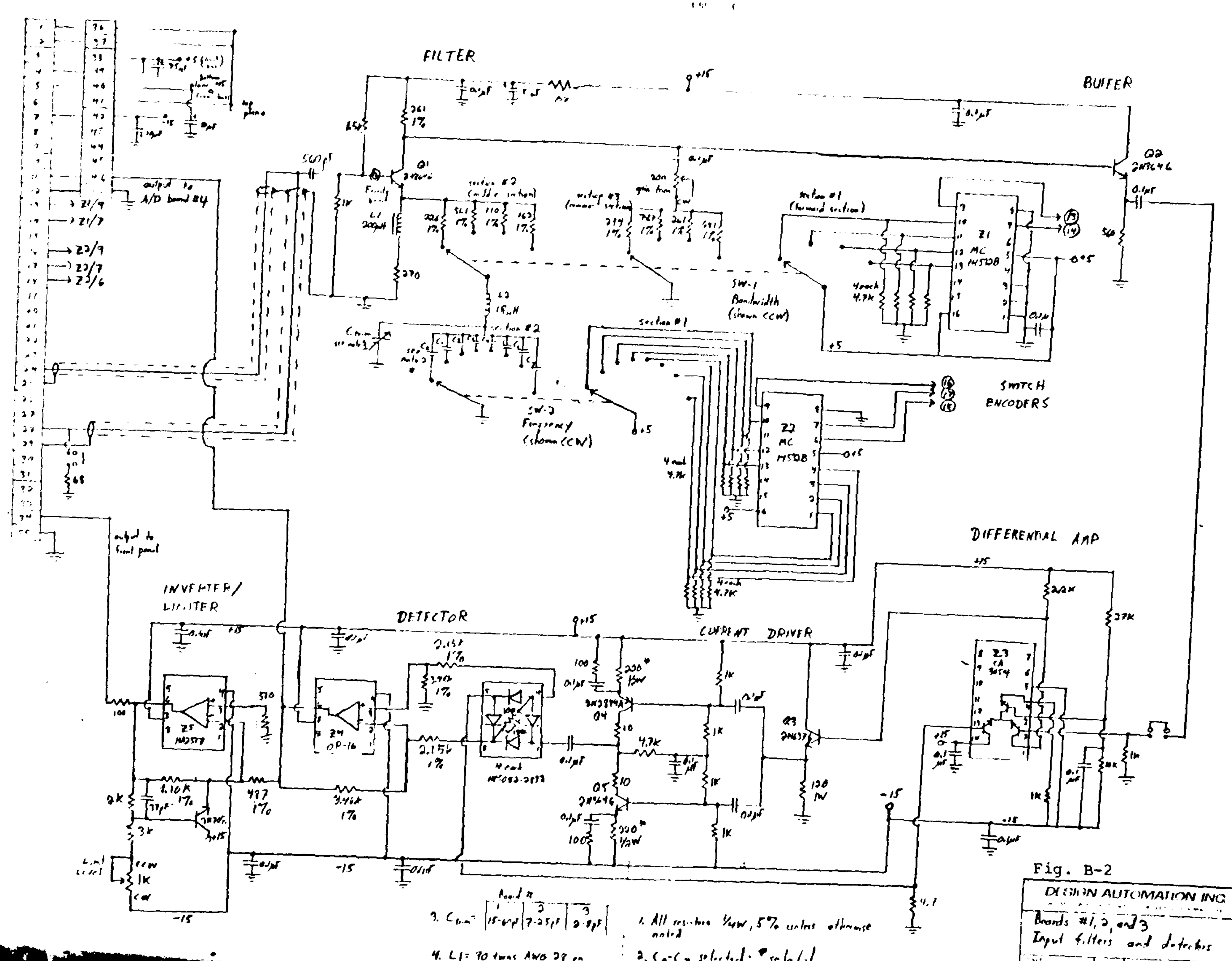

FILTER
BUFFER
SWITCH ENCODERS
DIFFERENTIAL AMP
CURRENT DRIVER
DETECTOR
INVERTER/ LIMITER
SW-1 Bandwidth (shown CCW)
SW-2 Frequency (shown CCW)
output to A/D board #4
output to front panel
1. All resistors 1/4W, 5% unless otherwise noted
2. C_a-C_g selected; * selected
3. C trim
4. L1 = 30 turns AWG 29 on Indiana General F624-19-Q1 toroid
DESIGN AUTOMATION INC
Boards #1, 2, and 3
Input filters and detectors

Fig. B-2

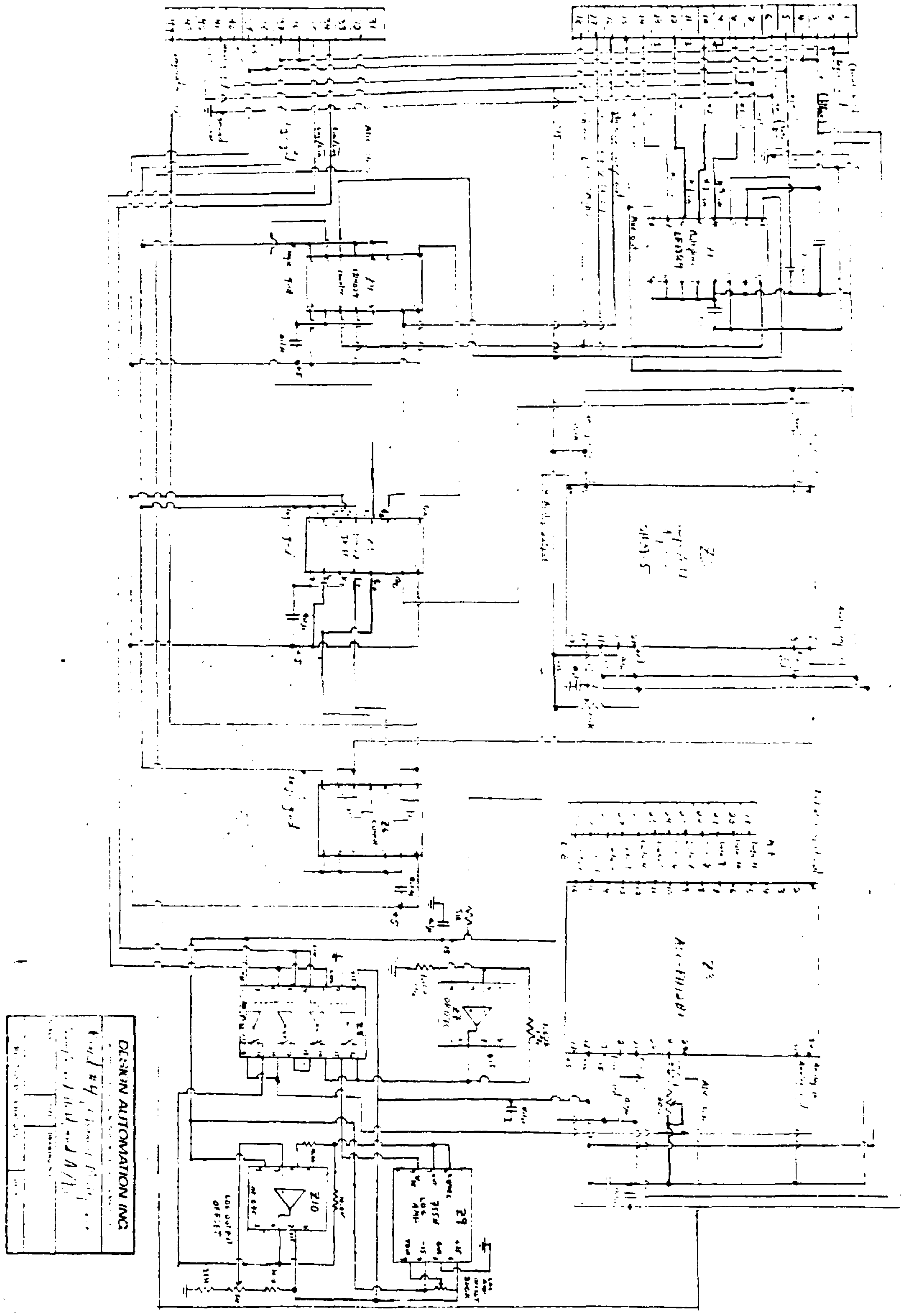


Fig. B-3

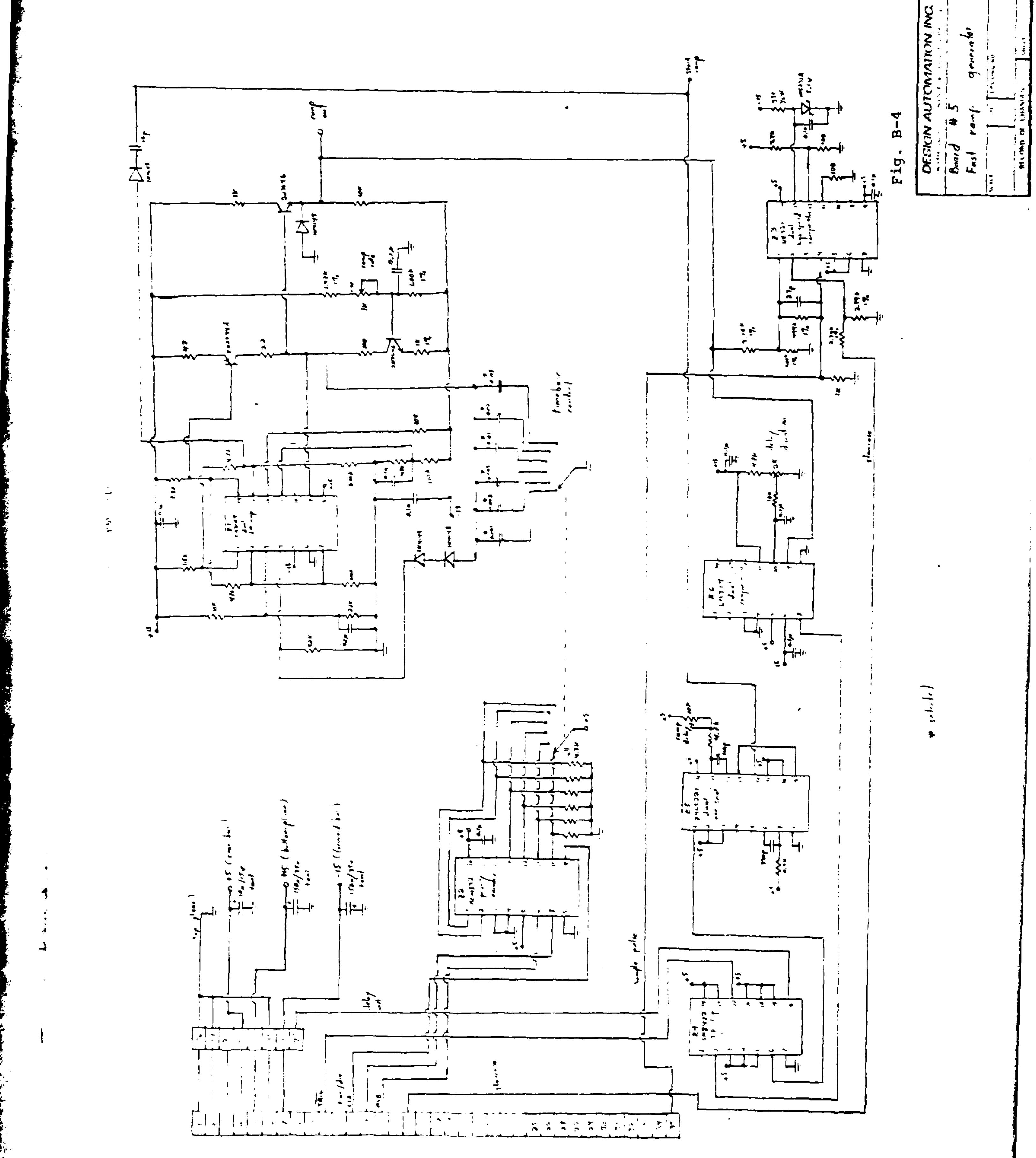


Fig. B-4

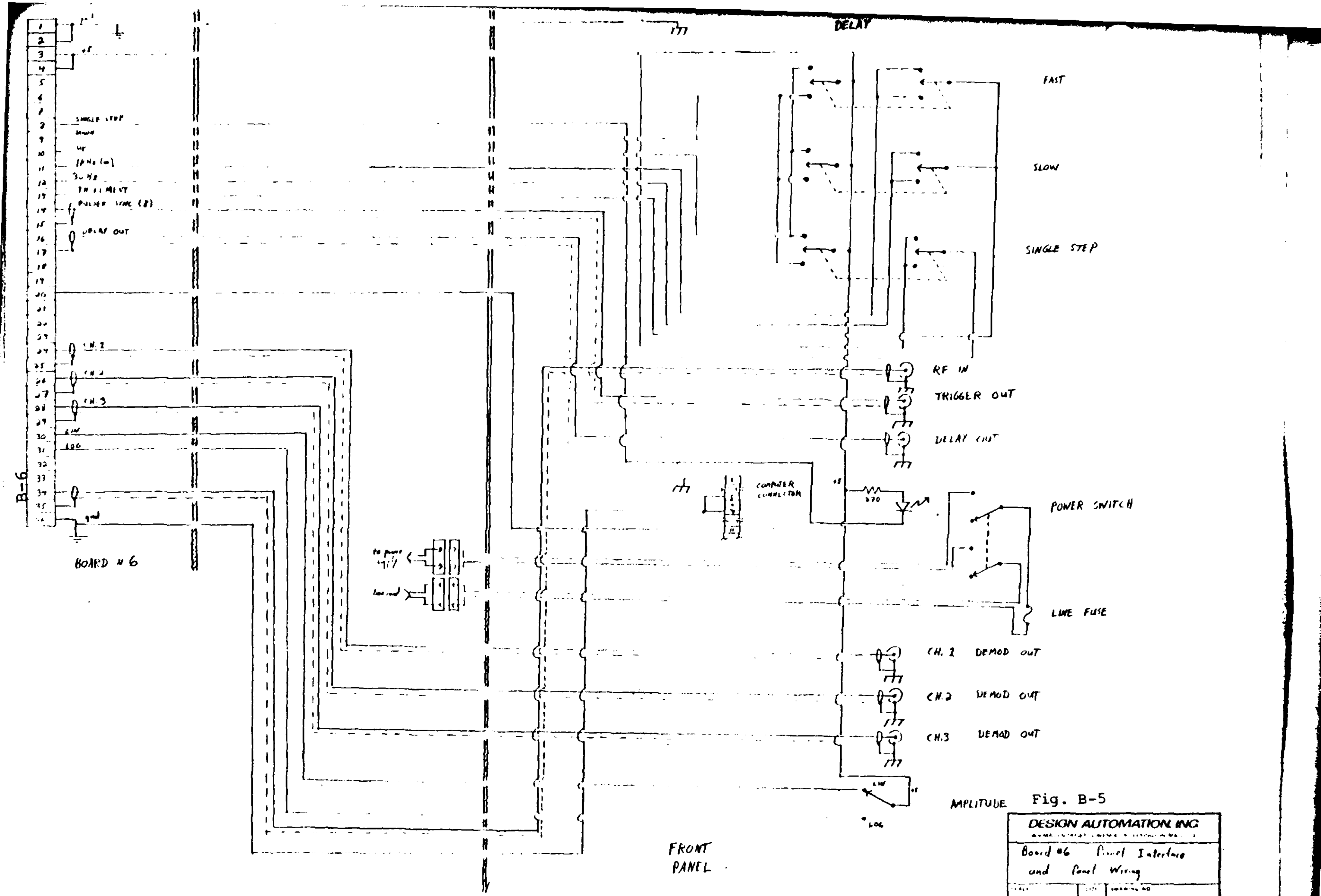


Fig. B-5

DESIGN AUTOMATION, INC.

Board #6 Panel Interface and Panel Wiring

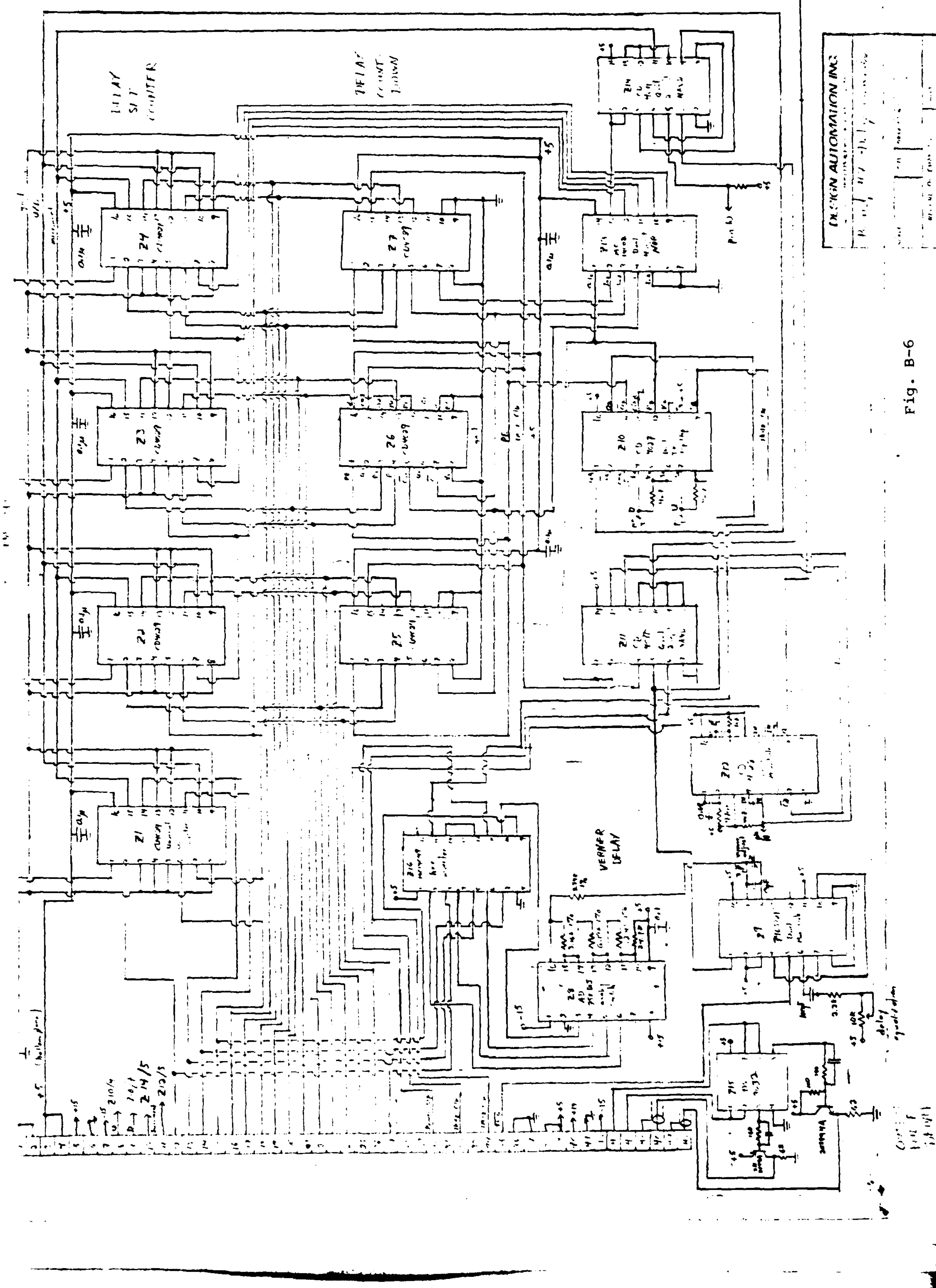


Fig. B-6

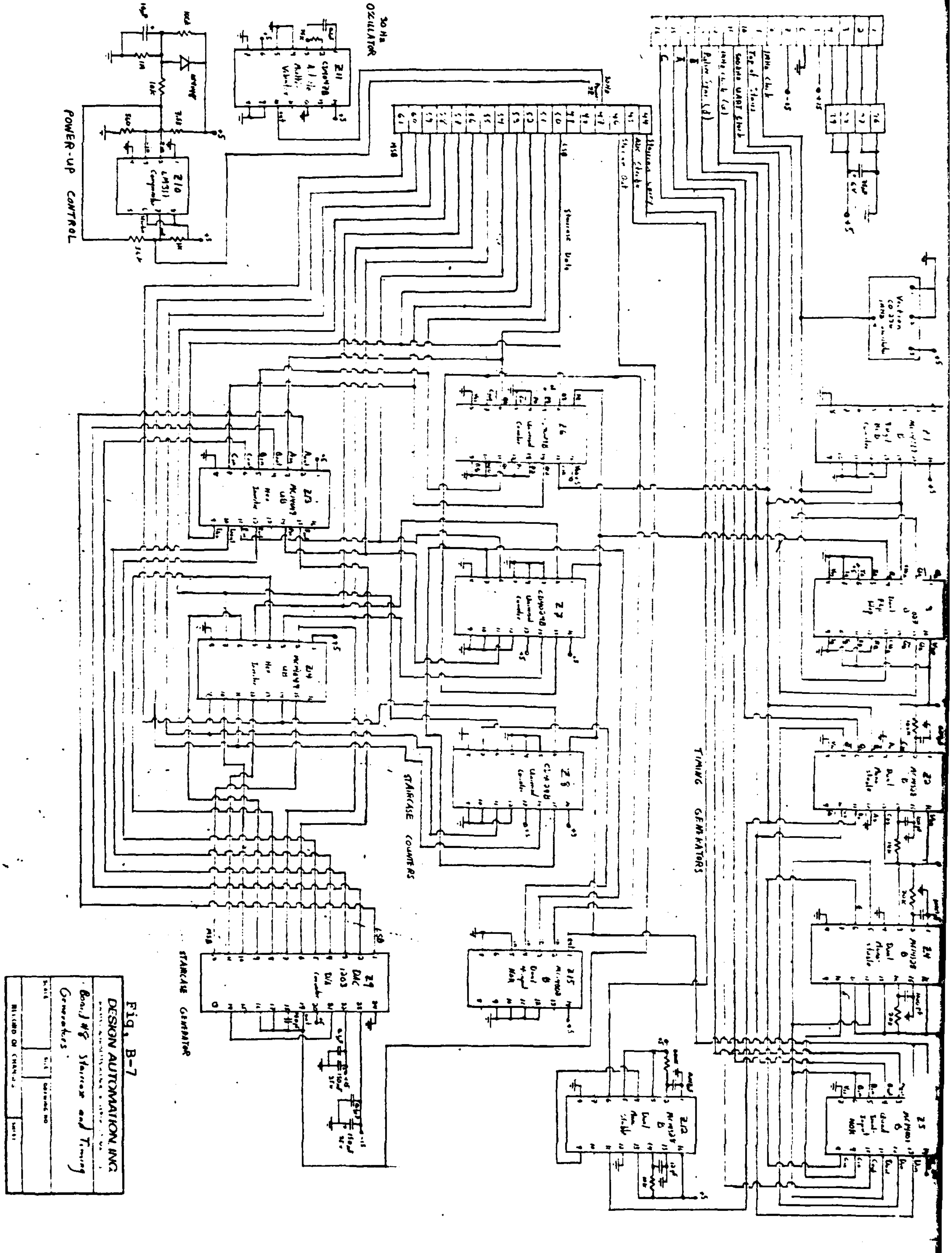
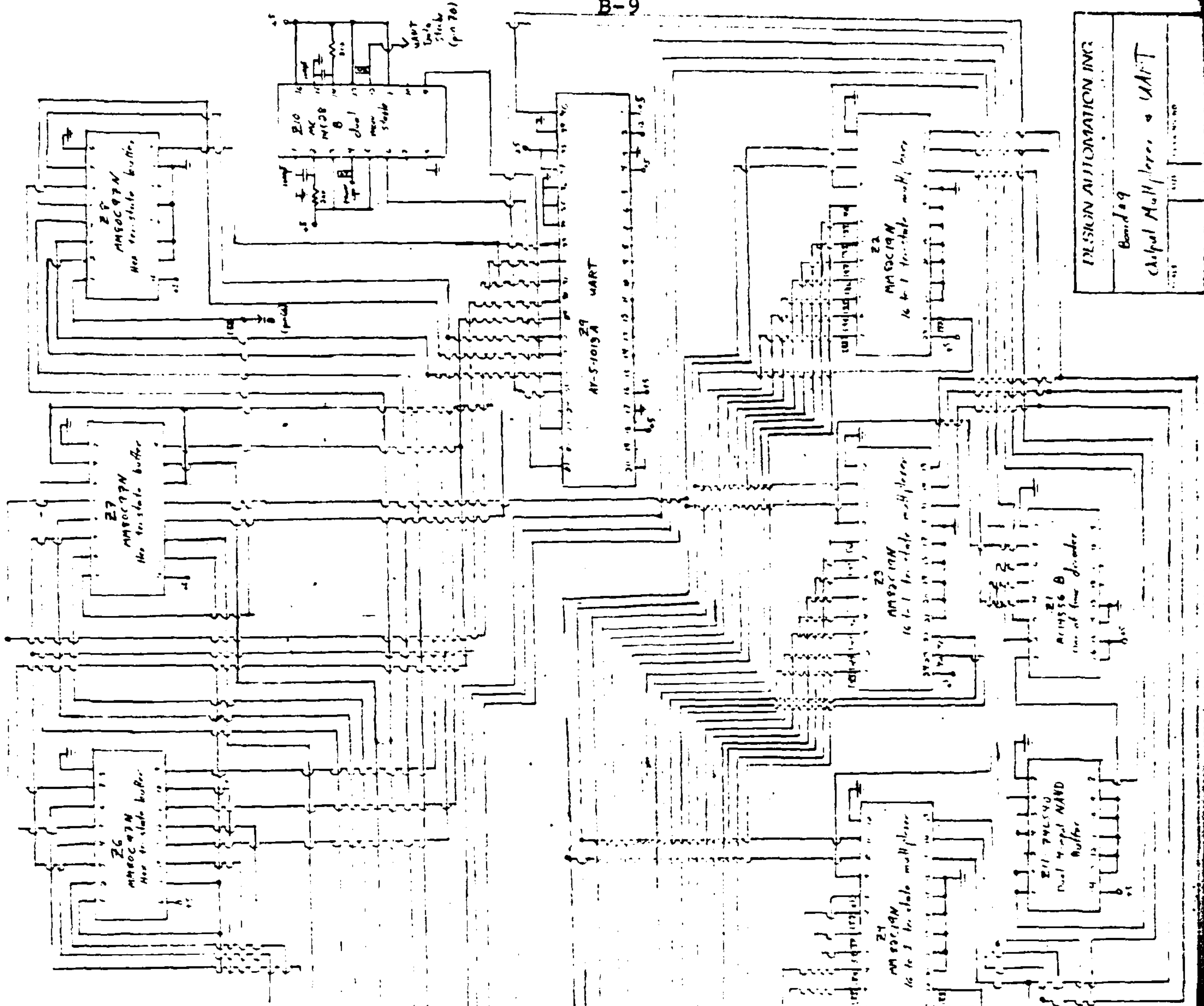

POWER-UP CONTROL
30 Hz OSCILLATOR
STAIRCASE COUNTERS
TIMING GENERATORS
STAIRCASE GENERATOR
DESIGN AUTOMATION, INC.
Board #8 Staircase and Timing Generators.

Fig. B-7


DESIGN AUTOMATION INC.
Board 9
Output Multiplexer & UART
Z9 AY-5-1013A UART
Z10 MC14538 B dual mono stable
Z8 MM80C97N Hex tri-state buffer
Z7 MM80C97N Hex tri-state buffer
Z6 MM80C97N Hex tri-state buffer
Z2 MM80C19N 16 to 1 tri-state multiplexer
Z3 MM80C19N 16 to 1 tri-state multiplexer
Z4 MM80C19N 16 to 1 tri-state multiplexer
Z5 MM80C19N 16 to 1 tri-state multiplexer
Z1 MC14556 B dual 4 line decoder
Z11 74LS40 Dual 4-input NAND buffer
UART data strobe
UART clock
4.7K

Power Supply Connections at Number 10 Connector:

Voltage	Pins											
Ground	1	2	6	9	10	14	16	17	21	23	24	28
	36	37	41	43	44	48	50	51	55	57	58	
+5V	3	4	11	12	18	19	25	26				
	38	39	45	46	52	53	59	60				
+15V	5	13	20	40	47	54	27					
−15V	7	15	22	42	49	56	29					
−12 V	8											

Fig. B-9

PAGE 1

TABLE B-I. BACKPLANE WIRING

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
1	1		10	1								A
1	2		10	2								A
1	3		10	3								A
1	4		10	4								A
1	5		10	5								A
1	6		10	6								A
1	7		10	7								A
1	13		9	60								A
1	14		9	61								A
1	16		9	51								A
1	17		9	52								A
1	18		9	53								A
1	36		10	36								A
1	37		10	37								A
1	38		10	38								A
1	39		10	39								A
1	40		10	40								A
1	41		10	41								A
1	42		10	42								A
1	46		4	8								A
1	47		4	9								
1	24		2	25								
1	25		2	29								
1	34		6	24								
1	35		6	25								

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STAT
2	1		10	1								A
2	2		10	2								A
2	3		10	3								A
2	4		10	4								A
2	5		10	5								A
2	6		10	6								A
2	7		10	7								A
2	13		9	62								A
2	14		9	63								A
2	16		9	54								A
2	17		9	55								A
2	18		9	56								A
2	36		10	36								A
2	37		10	37								A
2	38		10	38								A
2	39		10	39								A
2	40		10	40								A
2	41		10	41								A
2	42		10	42								A
2	46		4	10								A
7	47		4	11								
2	[illegible]		1	24								
2	[illegible]		1	25								
2	24		3	[illegible]								
			[illegible]	[illegible]								

PAGE 2

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
2	1		10	1								A
2	2		10	2								A
2	3		10	3								A
2	4		10	4								A
2	5		10	5								A
2	6		10	6								A
2	7		10	7								A
2	13		9	62								A
2	14		9	63								A
2	16		9	54								A
2	17		9	55								A
2	18		9	56								A
2	36		10	36								A
2	37		10	37								A
2	38		10	38								A
2	39		10	39								A
2	40		10	40								A
2	41		10	41								A
2	42		10	42								A
2	46		4	10								A
[illegible]	[illegible]		4	11								
[illegible]	[illegible]		1	[illegible]								
[illegible]	[illegible]		[illegible]	[illegible]								
[illegible]	[illegible]		[illegible]	[illegible]								
			[illegible]	[illegible]								

PAGE [illegible]

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]								
[illegible]	[illegible]		[illegible]	[illegible]								

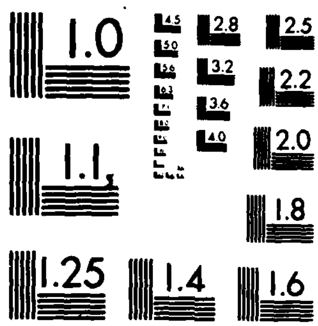
BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
3	1		10	9								A
3	2		10	10								A
3	3		10	11								A
3	4		10	12								A
3	5		10	13								A
3	6		10	14								A
3	7		10	15								A
3	13		9	64								A
3	14		9	65								A
3	16		9	57								A
3	17		9	58								A
3	18		9	59								A
3	36		10	43								A
3	37		10	44								A
3	38		10	45								A
3	39		10	46								A
3	40		10	47								A
3	41		10	48								A
3	42		10	49								A
3	46		4	12								A
[illegible]	[illegible]		[illegible]	[illegible]								
[illegible]	[illegible]	✓	[illegible]	[illegible]								
[illegible]	39	✓	[illegible]	[illegible]								
[illegible]	[illegible]	✓	[illegible]	[illegible]								
3	[illegible]	✓	[illegible]	[illegible]								

AD-A128 338 MULTIPLE-FREQUENCY ULTRASONIC PULSE-ECHO DISPLAY SYSTEM (U) DESIGN AUTOMATION INC LEXINGTON MASS 28 SEP 82 DAAG46-80-C-0010

22

UNCLASSIFIED F/G 9/2 NL

END
DATE FILMED
7-83
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

256	27				
2[illegible]6	29				

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
4	1		10	9								A
4	2		10	10								A
4	3		10	11								A
4	4		10	12								A
4	5		10	13								A
4	6		10	14								A
4	7		10	15								A
4	8		1	46								R
4	10		2	46								R
4	12		3	46								R
—												—
4	16		8	44								A
4	17		9	21								A
4	18		9	22								A
4	19		9	20								A
4	20		9	19								A
4	21		9	18								A
4	22		9	17								A
4	23		9	25								A
4	24		9	26								A
4	25		9	27								A
4	26		9	28								A
4	27		9	29								A
4	28		9	30								A
4	29		9	31								A

9	24					
8	45					
10	43					
10	44					
10	45					
10	46					
10	47					
10	48					
10	49					
5	35					
6	30			9	5	
6	31			9	7	
1	47					
2	47					
3	47					

6	1		10	16	
6	2		10	17	
6	3		10	18	
6	4		10	19	
10	20				
10	21				
10	22				
9	50				
9	49				
9	48				
8	46				
4	44				
6	36		10	50	
10	51				
10	52				
10	53				
10	54				
10	55				
10	56				
7	35				
7	43				

PAGE 7

rev 1

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
6	1		5	1		10	16					
6	2		5	2		10	17					
6	3		5	3		10	18					
6	4		5	4		10	19					
6	8		7	11								
6	9		7	8								
6	10		7	9								
6	11		7	12		8	12					
6	[illegible]		[illegible]	[illegible]								
[illegible]	13		7	[illegible]								
6	[illegible]		7	46		[illegible]						
6	[illegible]		7	45		[illegible]	coax					
6	16		7	48		[illegible]						
6	17		7	47		[illegible]	[illegible]					
6	23		1	[illegible]								
6	24		1	34								
6	25		1	35		shield	[illegible]					
6	[illegible]		2	[illegible]								
6	27		2	[illegible]		[illegible]						
6	[illegible]		[illegible]	[illegible]			[illegible]					
[illegible]	[illegible]		[illegible]	[illegible]		[illegible]						
6	[illegible]		[illegible]	[illegible]		10	50					
6	30		9	5		4	34					
6	31		9	7		4	35					
6	34		3	24								

6 [illegible] 3 [illegible]

10	18				
10	19				
10	20				
10	21				
10	22				
6	9				
6	10				
6	13				
9	44				
9	45				
9	46				
9	47				
9	32				
9	33				
9	34				
9	35				
9	36				

9	43				
8	48		9	69	
6	11		8	12	
8	9				
10	50				
10	51				
10	52				
10	53				
10	54				
10	55				
10	56				
8	13				
6	14				
[illegible]	6			[illegible]	
5	10				
6	15			[illegible]	
6	17				
6	8				
5	43				

10	23				
10	24				
10	25				
10	26				
7	34				
9	23				
9	67				
6	11		7	32	
~~6~~ 7	~~14~~ 44				
9	66				
9	16				
9	70				
10	57				
10	58				
10	59				
10	60				
4	16				
4	32				
5	16				
6	12				
7	30		9	69	
9	15				

10	27				
10	28				
10	29				

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
9	1		10	23								A
9	2		10	24								A
9	3		10	25								A
9	4		10	26								A
9	5		~~[illegible]~~			6	30		4	34		—
9	6		10	28		6	31		4	[illegible]		A
9	7		~~[illegible]~~									—
9	8		10	8								A
9	9		8	60								R
9	10		8	55								R
9	11		8	54								R
9	12		8	53								R
9	13		8	52								R
9	14		8	51								R
9	15		8	50								R
9	16		8	15								R
9	17		4	22								R
9	18		4	21								R
9	19		4	20								R
9	20		4	19								R
9	21		4	17								R
9	22		4	18								R
9	23		8	10								R
9	24		4	30								R
9	25		4	23								R

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
9	26		4	24								R
9	27		4	25								R
9	28		4	26								R
9	29		4	27								R
9	30		4	28								R
9	31		4	29								R
9	32		7	16								R
9	33		7	17								R
9	34		7	18								R
9	35		7	19								R
9	36		7	20								[illegible]
9	37		7	21								R
9	38		7	22								R
9	39		7	23								R
9	40		7	24								R
9	41		7	25								R
9	42		7	26								R
9	43		7	27								R
9	44		7	12								R
9	45		7	13								R
9	46		7	14								R
9	47		7	15								R
9	48		5	14								R
9	49		5	13								R
9	50		5	12								R

1	16				
1	17				
1	18				
2	16				
2	17				
2	18				
3	16				
3	17				
3	18				
1	13				
1	14				
2	13				
2	14				
3	13				
3	14				
8	14				
8	11				
6	20				
7	30		8	48	
8	16				

1	1		2	1	
1	2		2	2	
1	3		2	3	
1	4		2	4	
1	5		2	5	
1	6		2	6	
1	7		2	7	
9	8				
3	1		4	1	
3	2		4	2	
3	3		4	3	
3	4		4	4	
3	5		4	5	
3	6		4	6	
3	7		4	7	
5	1		6	1	
5	2		6	2	
5	3		6	3	
5	4		6	4	
5	5		7	5	
5	6		7	6	
5	7		7	7	
8	1		9	1	
8	2		9	2	
8	3		9	3	

8	4				
8	5				
8	6		9	6	
8	7				
1	36		2	36	
1	37		2	37	
1	38		12	38	
1	39		2	39	
1	40		2	40	
1	41		2	41	
1	42		2	42	
3	36		4	36	
3	37		4	37	
3	38		4	38	
3	39		4	39	
3	40		4	40	
3	41		4	41	
3	42		4	42	
5	36		6	36	

BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	BOARD	PIN	✓	STATUS
10	56		5	42		7	42					R
10	57		8	36								R
10	58		8	37								R
10	59		8	38								R
10	60		8	39								R

APPENDIX C.

OPERATING PROGRAM LISTINGS

1. KEE
2. SIMU
3. START
4. RETRIEVE
5. GRAPH
6. GRAFANEX

SECTION C-I. KEE

```
10 'This is PROGRAM KEE.
20  PRINT CHR$(12)
30 PRINT "~C7";"WAIT FOR `";"~C2";"?";"~C7";"' SYMBOL."
40 PRINT:PRINT "Respond, when ready, with ";"~C2";"CR";"~C7";"."
50 PRINT :PRINT
60 FOR K=2 TO 8
70 PRINT "~C5";"key F";K;":";"~C2"
80 FOR N=0 TO 63
90 H=PEEK(16384+64*(K-2)+N)
100 IF A=24 AND H=74 THEN H=10
110 A=0
120 IF H=24 THEN A=24
130 IF H=1 OR H=126 THEN GOTO 240
140 IF H=12 THEN GOTO 260
150 IF H=13 THEN PRINT CHR$(10);CHR$(13); :GOTO 180
160 IF H=27 THEN GOTO 280
170 PRINT CHR$(H);
180 NEXT
190 PRINT :PRINT
200 IF K<8 THEN INPUT Z$ ELSE PRINT
210 NEXT
220 PRINT :PRINT :PRINT
230 END
240 PRINT "~C7";"(";"~C2";"MODE";"~C7";")";"~C2";
250 GOTO 180
260 PRINT "~C7";"(";"~C2";"ERASE";"~C7";")";"~C2";
270 GOTO 180
280 PRINT "~C7";"(";"~C2";"ESC";"~C7";")";"~C2";
290 GOTO 180
```

SECTION C-2. SIMU

```
100 ' THIS IS PROGRAM SIMU
110 I0=256*PEEK(15175)+PEEK(15174) 'Buffer start address
120 ' Display Format
130 PRINT CHR$(12);"~U060255~C7";"Program  SIMU  - ";
140 PRINT " Preparation of simulated data"
150 PRINT "~G'";:PLOT 60,243,436,243:
160 PRINT "~K~P";CHR$(21)
170 PRINT CHR$(5);CHR$(5);CHR$(5);CHR$(5);SPC(20);"~C6";
180 PRINT "enter ";"~C5";"LIN";"~C6";" or ";"~C5";"LOG";"~C6";
190 INPUT SC$
200 IF SC$<>"LIN" AND SC$<>"LOG" THEN GOTO 130
210 PRINT CHR$(22);CHR$(22);CHR$(22);CHR$(22);
220 PRINT "~C6";"COMPUTE 3 FUNCTIONS, 448 points each:"
230 PRINT "~U123202~C2";"100";"~U223202";"200";
240 PRINT "~U323202";"300";"~U423202";"400";"~U471202";"448";
250 PRINT CHR$(14);"~U032190";"&";"~U132190";"&";"~U232190";"&";
260 PRINT "~U332190";"&";"~U432190";"&";"~U480190";"&";
270 PRINT CHR$(21);CHR$(15);"~C6"
280 PRINT
290 PRINT
300 PRINT "FORMAT 2688 BYTES:"
310 PRINT "~C2~U186137";"1000";"~U352137";"2000";
320 PRINT "~U467137";"2688";
330 PRINT CHR$(14);"~U032125";"&";"~U199125";"&";
340 PRINT "~U365125";"&";"~U480125";"&";
350 PRINT CHR$(21);CHR$(15);"~C6"
360 PRINT
370 PRINT
380 PRINT "LOAD 2688 MEMORY BINS:"
390 PRINT "~C2~U186072";"1000";"~U352072";"2000";
400 PRINT "~U467072";"2688";
410 PRINT CHR$(14);"~U032060";"&";"~U199060";"&";
420 PRINT "~U365060";"&";"~U480060";"&";
430 ' COMPUTE FUNCTIONS;  Amplitude must not exceed 2047!
440 DIM X1(448),X2(448),X3(448)
450 K=1600/LOG(160)
460 FOR N=0 TO 447
470 Y1=340*SIN((N+64)/40) + 400
480 Y2=800*COS(N/80)+800
490 Y3=370-6E-03*(N-200)^2
500 IF SC$="LIN" THEN GOTO 540
510 IF Y1>10 THEN Y1=K*LOG(Y1/10) ELSE Y1=0
520 IF Y2>10 THEN Y2=K*LOG(Y2/10) ELSE Y2=0
530 IF Y3>10 THEN Y3=K*LOG(Y3/10) ELSE Y3=0
540 X1(N)=(INT(Y1) MAX 0) MIN 2047
550 X2(N)=(INT(Y2) MAX 0) MIN 2047
560 X3(N)=(INT(Y3) MAX 0) MIN 2047
570 PRINT
580 PRINT "~C4~G%";:PLOT 35+N,178:
590 PRINT
600 NEXT
610 ' ASSEMBLE PARAMETER BITS
620 DIM S(34)
630 FOR K=0 TO 33
640 READ S(K)
```

PROGRAM SIMU CONTINUED

```
650 NEXT
660 DATA 0,0,1,1,0,0,1,0,0,1,0,1,0,1,0,0,0,1,1,0
670 DATA 0,1,0,1,1,1,1,0,0,0,0,1,1,1
680 IF SC$="LIN" THEN S(34)=0 ELSE S(34)=1
690 ' FORMAT BYTES
700 DIM A(1344),B(1344)
710 FOR I=0 TO 447
720 IF I=0 THEN D1=1 ELSE D1=0
730 IF I>34 THEN D2=0 ELSE D2=1-S(I)
740 P=INT(X1(I)/128)
750 A(I)=D1*128+D2*64+P
760 B(I)=X1(I)-128*P
770 IN=35+I/3
780 PRINT "~C4~G%";:PLOT IN,113:
790 NEXT
800 PRINT 447
810 FOR I=448 TO 895
820 IF I=448 THEN D1=1 ELSE D1=0
830 IF I>481 THEN D2=0 ELSE D2=S(I-448)
840 A(I)=128*D1+64*D2+16+INT((X2(I-448))/128)
850 B(I)= X2(I-448)-128*INT((X2(I-448))/128)
860 IN=35+I/3
870 PRINT "~C4~G%";:PLOT IN,113:
880 NEXT
890 FOR I=896 TO 1343
900 IF I=896 THEN D1=1 ELSE D1=0
910 IF I>929 THEN D2=0 ELSE D2=S(I-896)
920 A(I)=128*D1+64*D2+32+INT((X3(I-896))/128)
930 B(I)=X3(I-896)-128*INT((X3(I-896))/128)
940 IN=35+I/3
950 PRINT "~C4~G%";:PLOT IN,113:
960 NEXT
970 ' LOAD MEMORY BINS
980 FOR I=0 TO 1343
990 POKE (I0+100+2*I),(A(I))
1000 POKE (I0+100+2*I+1),(B(I))
1010 IN=35+I/3
1020 PRINT "~C4~G%";:PLOT IN,48:
1030 NEXT
1040 FOR N=0 TO 99 :POKE I0+N,0 :NEXT
1050 POKE I0+19,6
1060 POKE I0+21,1
1070 POKE I0+8,1
1080 POKE I0+9,1
1090 POKE I0+10,1
1100 POKE I0+11,3
1110 POKE I0+12,232
1120 PRINT
1130 PRINT
1140 PRINT CHR$(21);CHR$(15);"~C7~P";
1150 PRINT "DATA IN BUFFER BEGINNING WITH ";I0
1160 PRINT "SIMU has been completed.";
1170 END
```

SECTION C-3. START

```
100 'THIS IS PROGRAM START
110 DA=256*PEEK(15175)+PEEK(15174)
120 POKE DA+21,1
130 'Load SNAPSHOT routine
140 DIM AR$(46)
150 FOR N=0 TO 45
160 READ AR$(N)
170 POKE 17000+N,VAL("&H"+AR$(N))
180 NEXT
190 DATA 2A,46,3B,01,64,00,09,01,00,15,1E,32,DB,4D,CB,4F
195 DATA CA,8B,42,DB,4C,77,23,0B,3E,00,B8,C2,72,42,B9
200 DATA C2,72,42,C9,1D,3E,00,BB,C2,74,42,32,CC,42,C9
240 PRINT CHR$(27);"OA0";CHR$(21);CHR$(12);"~N~C7~K";
250 PRINT "THIS BEGINS THE DATA DISPLAY AND PROCESSING ";
260 PRINT "PROGRAMS."  :PRINT :PRINT
270 PRINT "Subsequent steps are automatic, or cued for ";
280 PRINT "operator response." :PRINT
290 PRINT "There are two kinds of discretionary response:":PRINT
300 PRINT SPC(10);"First,  if the cue reads ";"~C6";"press";
310 PRINT "~C7";", then only a" :PRINT SPC(18);
320 PRINT "single key is involved.  Press it and"
330 PRINT SPC(18);"wait for response.":PRINT :PRINT SPC(10);
340 PRINT "Second, if the cue reads ";"~C6";"type";"~C7";
350 PRINT ", or calls for numerical" :PRINT SPC(18);
360 PRINT "entry, several keys must be pressed in"
370 PRINT SPC(18);"sequence.  Follow these with a"
380 PRINT SPC(18);"~C6";"CARRIAGE RETURN";
390 PRINT "~C7";", (";"~C5";"CR";"~C7";")."
400 FOR N=1 TO 8 :PRINT :NEXT :PRINT SPC(35);
410 GOTO 430
420 PRINT CHR$(12);"~C7";
430 X$=""
440 PRINT "To continue, press ";"~C5";"CR";"~C7";".";"~C0";
450 INPUT X$
460 IF X$<>"" THEN GOTO 420
470 PRINT CHR$(12);"~C7";
480 PRINT "THERE ARE THREE SOURCES OF DATA FOR DISPLAY:"
490 PRINT  :PRINT  :PRINT SPC(10);"1.  The ";"~C6";
500 PRINT "FILTER-DIGITIZER";"~C7";".  Data from this unit"
510 PRINT SPC(14);"are organized into groups called ";"~C6";
520 PRINT "SNAPSHOTS";"~C7";"." :PRINT SPC(14);"These comprise";
530 PRINT " about 2 seconds of the unit's" :PRINT SPC(14);
540 PRINT "output, entered into buffer memory on command."
550 PRINT :PRINT SPC(10);"2.  ";"~C6";"DISK FILE";"~C7";
560 PRINT ", (Disk Drive #2).  Previous SNAPSHOTS,"
570 PRINT SPC(14);"processed perhaps, can be reentered into ";
580 PRINT "buffer" :PRINT SPC(14);"memory for display."
590 PRINT :PRINT SPC(10);"3.  Information left over in buffer ";
600 PRINT "memory from" :PRINT SPC(14);"a previous operation."
610 PRINT :PRINT :PRINT "One may select from these sources ";
620 PRINT "as follows:"
630 PRINT :PRINT :PRINT :PRINT :PRINT SPC(35);
640 GOTO 660
650 PRINT CHR$(12);"~C7";
660 X$=""
670 PRINT "(To continue, press ";"~C5";"CR";"~C7";".)";"~C0";
680 INPUT X$
690 IF X$<>"" THEN GOTO 650
700 PRINT CHR$(12);"~C6"; :PRINT :PRINT
710 PRINT "To call for a SNAPSHOT, press key ";
```

PROGRAM START CONTINUED

```
720 PRINT "~C5";"F4";"~C6";"."  :PRINT :PRINT
730 PRINT "To recall data in disk file, press key ";
740 PRINT "~C5";"F5";"~C6";"."  :PRINT :PRINT
750 PRINT "To display data already in buffer memory, ";
760 PRINT "press key ";"~C5";"F6";"~C6";"."
770 PRINT :PRINT :PRINT :PRINT :PRINT
780 DOS"LOAD GRAPH/1"
```

SECTION C-4. RETRIEVE

```
100 'PROGRAM RETRIEVE
110 CLEAR 4000
120 DIM A$(100),B$(100),C$(10)
130 ON ERROR #0 GOTO 0
140 PRINT CHR$(27);"OA1";"~W152195344085~R";
150 PRINT CHR$(27);"OA0";
160 PRINT CHR$(12);"~C7";"TO RETRIEVE A RECORD FROM DISKFILE:"
170 PRINT :PRINT SPC(10);"First  - Load the appropriate disk ";
180 PRINT "in Drive #2." :PRINT
190 K=0  :Y=0
200 ON ERROR GOTO 230
210 DOS"LDDIR /2 A$"
220 GOTO 240
230 RESUME
240 PRINT "~N~C2~U200215";"Directory"
250 B$(0)="************************"
260 B$(2)=B$(0)
270 B$(1)=A$(0)
280 FOR N=3 TO 100
290 IF LEFT$(A$(N-2),1)="" THEN GOTO 320
300 B$(N)=A$(N-2)
310 GOTO 350
320 F=N-3 MAX 8
330 Y=(N-3) MIN 3
340 GOTO 360
350 NEXT
360 IF F=0 THEN GOTO 400
370 PRINT "~C4~U";
380 PLOT 125,(174-10*Y)
390 PRINT CHR$(14);"++%";CHR$(15);"~C2~K";
400 GOSUB 650
410 PRINT CHR$(27);"OA0";"~U080070~C7";"Second - Scroll the ";
420 PRINT "Directory up or down until the";"~U152060";
430 PRINT "red arrow points to the desired File#.";"~U152050";
440 PRINT "Use the gray cursor controls, ";"~C5";CHR$(14);"'";
450 PRINT "~C7";CHR$(15);" and ";"~C5";CHR$(14);"&";
460 PRINT "~C7";CHR$(15);".";
470 PRINT "~?020~U080030";"Third  - When ready, ";
480 PRINT "press ";"~C5";"L";"~C7";" to load the file, - or"
490 PRINT "~U152020";"select another disk and press ";"~C5";"R";
500 PRINT "~C7";" for repeat, -";"~U152010";
510 PRINT "or press ";"~C5";"CR";"~C7";" to exit the program.";
520 ON ERROR #1 GOTO 540
530 GOTO 530
540 IF ERR=24 THEN Q=INP(&H4A) ELSE ON ERROR #0 GOTO 0
550 IF Q=76 OR Q=82 OR Q=13 THEN RESUME 590
560 IF Q=11 THEN GOSUB 730
570 IF Q=10 THEN GOSUB 820
580 RESUME
590 IF Q=82 GOTO 130  '(repeat)
```

PROGRAM RETRIEVE CONTINUED

```
600 IF Q=13 GOTO 1180 '(exit)
610 IF Q=76 GOTO 1000
620 PRINT CHR$(12);"~C4";"ERROR - START AGAIN!   Press ";
630 PRINT "key ";"~C5";"F5";"~C4";"."
640 END
650 'DIRECTORY PRINTOUT subroutine
660 PRINT CHR$(27);"OA1";"~K";CHR$(21);
670 FOR N=0 TO 10
680 IF N=0 OR N=2 THEN PRINT"~N~C6";ELSE PRINT"~N~C2";
690 IF N<10 THEN PRINT B$(N) ELSE PRINT B$(N);
700 ON ERROR GOTO 0
710 NEXT
720 RETURN
730 'UP ONE LINE subroutine
740 PRINT CHR$(27);"OA1";
750 PRINT "~U152095";CHR$(10);
760 IF K+12>0 THEN L=(K+11)MOD(F+3)
770 IF K+12<=0 THEN L=F+2-ABS(K+12)MOD(F+3)
780 IF L=0 OR L=2 THEN PRINT "~C6"; ELSE PRINT "~C2";
790 PRINT B$(L);
800 K=K+1
810 RETURN
820 'DOWN ONE LINE subroutine
830 PRINT CHR$(27);"OA1";
840 PRINT "~U152195"; CHR$(11);
850 IF K>0 THEN L=(K-1)MOD(F+3) ELSE L=F+2-ABS(K)MOD(F+3)
860 IF L=0 OR L=2 THEN PRINT "~C6"; ELSE PRINT "~C2";
870 PRINT B$(L);
880 K=K-1
890 RETURN
900 'WRONG SELECTION subroutine
910 PRINT CHR$(27);"OA0";"~P";
920 ON ERROR #0 GOTO 0
930 PRINT "~U380165~C4";"Select ";"~C2";".ABS ";"~C4"
940 PRINT "~U380145";"files only!"
950 FOR N=1 TO 400 :PRINT CHR$(7) :NEXT
960 PRINT "~?020";
970 PRINT "~U380165~C0";"              ";"~C7";
980 PRINT "~U380145";"              "
990 RETURN
1000 'LOAD sequence
1010 M=(F+2) MAX 8
1020 R=K+Y+2
1030 IF R=>0 THEN L=R MOD (M+1) ELSE L=M-(ABS(R+1) MOD (M+1))
1040 P$=LEFT$(B$(L),8)
1050 Q$=MID$(B$(L),10,3)
1060 IF Q$="ABS" GOTO 1090
1070 GOSUB 900
1080 GOTO 520
1090 DA=256*PEEK(15175)+PEEK(15174)
```

PROGRAM RETRIEVE CONTINUED

```
1100 DOS"FETCH "+P$+"/2 "+HEX$(DA)+" "
1110 PRINT CHR$(27);"OA0";"~X002~Y002~C6~U380165";"Wait"
1120 PRINT "~X001~Y001"
1130 POKE DA+19,5
1135 POKE DA+21,OT
1140 ERASE A$,B$,C$
1150 CLEAR
1160 ON ERROR #0 GOTO 0
1170 DOS"CHAIN GRAPH/1"
1180 PRINT CHR$(27);"OA0";CHR$(12);"~C6"; :PRINT :PRINT
1190 PRINT "To call for a new SNAPSHOT, press key ";
1200 PRINT "~C5";"F4";"~C6";"."  :PRINT :PRINT
1210 PRINT "To recall data in disk file, press key ";
1220 PRINT "~C5";"F5";"~C6";"."  :PRINT :PRINT
1230 PRINT "To display data already in buffer memory, ";
1240 PRINT "press key ";"~C5";"F6";"~C6";"."
1250 PRINT  :PRINT  :PRINT
1260 END
1270 RESUME 1280
1280 'Disk error trap sequence
1290 NF=0  :E$=""  :PRINT "~R"
1300 IF ERR=70 GOTO 1360
1310 IF ERR=71 GOTO 1370
1320 IF ERR=82 GOTO 1380
1330 IF ERR=83 GOTO 1390
1340 IF ERR=119 GOTO 1400
1350 GOTO 1410
1360 E$="#14."  :GOTO 1430
1370 E$="#15."  :GOTO 1430
1380 E$="#20."  :GOTO 1430
1390 E$="#21."  :GOTO 1430
1400 E$="#45."  :GOTO 1430
1410 E$="AS NOTED BELOW."
1420 EF=1
1430 PRINT CHR$(12);"~C4";
1440 F1$=LEFT$(P$,4)+"-"+RIGHT$(P$,4)
1450 PRINT "DISK SYSTEM HAS REJECTED REQUEST ";
1460 PRINT "TO LOAD #";F1$;","
1470 PRINT "CLAIMING DOS ERROR ";E$  :PRINT
1480 PRINT "~C6";"Correct problem and type ";"~C5";"GO";
1490 PRINT"~C6";". To exit program, press ";"~C5";"CR";"~C6";"."
1500 IF EF=1 THEN ON ERROR #0 GOTO 0
1510 K$=""
1520 INPUT K$
1530 IF K$="" THEN GOTO 1180
1540 IF K$="GO" THEN GOTO 130
1550 PRINT
1560 GOTO 1480
```

SECTION C-5. GRAPH

```
100  'THIS IS PROGRAM GRAPH
110  '
120 CLEAR 1000
130 ON ERROR #0 GOTO 0
140 DEFINT C,I-N,R-S,X
150 DIM SD(4),S(35),X0(448),X1(448),X2(448),V$(0)
160 READ A1,A2,A3,D1,D2,D3,F1,F2,F3,B1,B2,B3,ST,T,DEL,IVL,SC,NM
170 DATA 1,1,1,0,0,0,0,0,0,0,0,0,22,.125,-1,0,0,200
180 'DEFINE 3 WINDOWS
190 PRINT CHR$(27);"OA1";"~W065056511255~N~C1~K~P";
200 PRINT CHR$(27);"OA2";"~W417244502254~M~C4~N~C3~K~P";
210 PRINT CHR$(27);"OA3";"~W070032503042~M~C6~N~C4~K~P";
220 PRINT CHR$(27);"OA0";"~K~P~N~C7";CHR$(21);
230 DA=256*PEEK(15175)+PEEK(15174)
240  '
250 LF=PEEK(DA+19)
260 IF LF=4 THEN GOTO 690
270 IF LF=5 OR LF=6 THEN GOSUB 5020 '(retrieve stored param)
280 IF LF=5 THEN GOTO 340
290 IF LF=6 THEN PRINT CHR$(12)  :GOTO 730
300 PRINT CHR$(12);"~C4";"ERROR: entry code = ";LF;"~C7":PRINT
310 INPUT "ENTER CORRECT CODE (4, 5, or 6): ";LF
320 GOTO 260
330  '
340 ' File Load Message
350 PRINT CHR$(12);"FILE # ";F1$;" has been loaded." :PRINT
360 PRINT "Do you wish a QUICK PLOT? -- Press ";"~C5";"CR";"~C7"
370 PRINT SPC(14);"(scale modification possible)"
380 PRINT
390 PRINT SPC(9);"or a FULL PLOT? -- Type ";"~C5";"FULL";"~C7"
400 PRINT SPC(14);"(file no. modification possible)"
410 PRINT
420 O$=""
430 INPUT O$
440 PRINT"~C6";"WAIT;  start address search now in process.";
450 PRINT "~?030"
460 GOSUB 2930  '(locate start address)
470 IF O$<>"FULL" GOTO 820
480 IF PEEK(DA+21)=1 THEN GOSUB 4840 '(one-time msg)
490 GOSUB 1080 '(axes)
500 IF PEEK(DA+20)=1 THEN GOSUB 4790 '(print file #)
```

```
510 GOSUB 1940  '(full plot)
520  '
530 ' Full Plot End Message
540 GOSUB 4610  '(say when ready subroutine)
550 PRINT CHR$(27);"OA3";CHR$(12);
560 PRINT "~M~C4";"Type ";"~C5";"FILE";
570 PRINT "~C4";" or ";"~C5";"QUICK";"~C4";" or if done, press";
580 PRINT "~C5";" CR     ";"~C4";
590 P$=""
600 INPUT P$
610 IF P$<>"" AND P$<>"QUICK" AND P$<>"FILE" THEN GOTO 530
620 PRINT "~M~C0";CHR$(12);"~M~C6";
630 PRINT CHR$(27);"OA0";CHR$(21);"~U236042~C7";"microseconds"
640 IF P$="QUICK" THEN GOSUB 2280  '(graph lines)
650 IF P$="QUICK" THEN GOTO 850  '(print FILE #; QUICK plot)
660 IF P$="FILE" THEN GOSUB 4240  '(assign FILE #)
670 DOS"CHAIN GRAFANEX/1"  '(param storage and final message)
680  '
690 ' Snapshot Message
700 PRINT CHR$(12);"~C7";"A NEW SNAPSHOT HAS BEEN TAKEN."
710 POKE DA+20,0
720 PRINT
730 PRINT "A QUICK PLOT, which permits scale modification,"
740 PRINT "will follow in a moment."
750 PRINT
760 PRINT "Later, a FULL PLOT may be used for permanent filing "
770 PRINT "of the data."
780 PRINT "~?060";:PRINT
790 PRINT"~C6";"WAIT;  start address search now in process.";
800 PRINT "~?030"
810 GOSUB 2930  '(locate start address)
820 GOSUB 3070  '(param calc)
830 IF PEEK(DA+20)=1 THEN SC=PEEK(DA+27)
840 GOSUB 1080  '(axes)
850 IF PEEK(DA+20)=1 THEN GOSUB 4790  '(print file #)
860 GOSUB 3280  '(quick plot)
870 IF SC=0 THEN GOSUB 4050 ELSE GOSUB 5340 '(overall scale chg)
880 IF PP=1 THEN GOSUB 2280  ' (graph lines)
890 IF PB=1 AND SC>0 THEN ON SC GOSUB 1620,1680,1740,1800
900                '(vertical scales)
910 IF PB=1 THEN GOTO 850   ' (print FILE#; QUICK plot)
920 IF SC=0 THEN GOSUB 3600 ELSE GOSUB 5470   '(A's or D's)
930 IF PA+PB=0 THEN GOTO 970 'Implies no scale factor change
940 GOSUB 2280  ' (graph lines)
950 GOTO 850   ' (print FILE#; QUICK plot)
960  '
970 ' Quick Plot End Message
980 PRINT CHR$(21);CHR$(27);"OA3";CHR$(12);
990 PRINT "~M~C4";"Full Plot? - Type ";"~C5";
1000 PRINT "FULL";"~C4";"   or done? - Press ";"~C5";"CR    ";
```

```
1010 O$=""
1020 INPUT O$
1030 PRINT "~M~C0";CHR$(12);"~M~C6";
1040 PRINT CHR$(27);"0A0";CHR$(21);"~U236042~C7";"microseconds"
1050 IF O$="FULL" GOTO 480
1060 IF O$="" GOTO 670 ELSE GOTO 970
1070  '
1080 PRINT CHR$(12); ' AXES  subroutine
1090 PRINT "~G'~K~P~C7";:PLOT 64,55,64,255:
1100 PLOT 64,55,511,55
1110 ' vert axis label
1120 PRINT CHR$(21);
1130 PRINT "~C7~V~U004199";"RELATIVE";"~U024205";"AMPLITUDE"
1140 IF SC>0 THEN PRINT "~H~U002109";"(db)" :GOTO 1160
1150 PRINT "~H~U004109";"(%)"
1160 ON SC+1 GOSUB 1570,1620,1680,1740,1800  '(vert scales)
1170 ' horiz axis label
1180 PRINT "~U000030";"display";
1190 PRINT "~U000020";"begins at";
1200 PRINT "~U000010";"microseconds";
1210 PRINT "~U236042";"microseconds";
1220 ' "start arrow"
1230 PRINT CHR$(14);"~U061053";"#";"~U061044";"&"
1240 PRINT "~U061036";"8";"~U074020";"-III,";CHR$(15);
1250 GOSUB 2280  '(graph lines)
1260 ' variable parameters
1270 PRINT "~U000020";"begins at ";
1280 IF INT(DEL)>-1 AND INT(DEL)<10 THEN PRINT USING "#.# ";DEL;
1290 IF INT(DEL)>9 AND INT(DEL)<100 THEN PRINT USING "##.#";DEL;
1300 IF INT(DEL)>99 THEN PRINT USING "###.#";DEL;
1310 PRINT "~C2~U152030";
1320 IF SC=0 THEN PRINT "A= ";USING "##.#####";A1  :GOTO 1340
1330 PRINT "D=  ";USING "+## db   ";D1
1340 PRINT "~U152020";"f= ";
1350 IF F1>0 THEN PRINT USING "##.## Mhz";F1
1360 PRINT "~U144010";"bw= ";
1370 IF B1>0 THEN PRINT USING "##.## Mhz";B1
1380 PRINT "~C4~U280030";
1390 IF SC=0 THEN PRINT "A= ";USING "##.#####";A2 :GOTO 1410
1400 PRINT "D=  ";USING "+## db   ";D2
1410 PRINT "~U280020";"f= ";
1420 IF F2>0 THEN PRINT USING "##.## Mhz";F2
1430 PRINT "~U272010";"bw= ";
1440 IF B2>0 THEN PRINT USING "##.## Mhz";B2
1450 PRINT "~C3~U416030";
1460 IF SC=0 THEN PRINT "A= ";USING "##.#####";A3 :GOTO 1480
1470 PRINT                    "D=  ";USING "+## db   ";D3
1480 PRINT "~U416020";"f= ";
1490 IF F3>0 THEN PRINT USING "##.## Mhz";F3
1500 PRINT "~U408010";"bw= ";
```

```
1510 IF B3>0 THEN PRINT USING "##.## Mhz";B3
1520 'horiz axis scale
1530 ON (IVL+1) GOSUB 2490,2570,2650,2720,2790,2860
1540 IF LF=5 THEN GOSUB 4790  '(print FILE# subroutine)
1550 RETURN
1560  '
1570 'VERT AXIS LINEAR subroutine
1580 PRINT "~U031255";"100";"~U041215";"80"
1590 PRINT "~U041175";"60";"~U041135";"40"
1600 PRINT "~U041095";"20";"~U051059";"0"
1610 RETURN
1620 'VERT AXIS LOG 1-DECADE subroutine
1630 GOSUB 1850  '(erase vert scale)
1640 PRINT "~U051255";"0";"~U043219";"-4"
1650 PRINT "~U043179";"-8";"~U039139";"-12"
1660 PRINT "~U039099";"-16";"~U035059";"-20"
1670 RETURN
1680 'VERT AXIS LOG 2-DECADE subroutine
1690 GOSUB 1850  '(erase vert scale)
1700 PRINT "~U051255";"0";"~U043219";"-8"
1710 PRINT "~U039179";"-16";"~U039139";"-24"
1720 PRINT "~U039099";"-32";"~U035059";"-40"
1730 RETURN
1740 'VERT AXIS LOG 3-DECADE subroutine
1750 GOSUB 1850  '(erase vert scale)
1760 PRINT "~U051253";"0";"~U039224";"-10";"~U039191";"-20"
1770 PRINT "~U039158";"-30";"~U039125";"-40"
1780 PRINT "~U039092";"-50";"~U035059";"-60"
1790 RETURN
1800 'VERT AXIS LOG 4-DECADE subroutine
1810 GOSUB 1850  '(erase vert scale)
1820 PRINT "~U051255";"0";"~U039209";"-20"
1830 PRINT "~U039159";"-40";"~U039109";"-60";"~U035059";"-80"
1840 RETURN
1850 'ERASE VERT SCALE subroutine
1860 PRINT"~V~M~C0~X004~Y002~U032255";
1870 FOR N=0 TO 9
1880 PRINT CHR$(32);
1890 NEXT
1900 PRINT "~U028059";CHR$(32);
1910 PRINT "~H~N~C7~X001~Y001";
1920 RETURN
1930  '
1940 'FULL PLOT subroutine
1950 IF PEEK(DA+21)=1 THEN GOSUB 4840 '(one-time msg)
1960 POKE DA+21,0
1970 FOR I=0 TO 446
1980 FOR K=0 TO 2
1990 P=Q0+K*894+2*I+2
2000 NA=PEEK(P)
```

5

```
2010 NB=PEEK(P+1)
2020 CH=(NA AND 48)/16
2030 XI=(NA AND 15)*128+NB
2040 ON CH+1 GOTO 2050,2110,2170
2050 IF SC=0 THEN X0(I)=(XI*T/A1 MIN 199) MAX 1 :GOTO 2080
2060 XJ=MM*(1-(4/SC)*(1+(D1/80)-(XI/1600)))
2070 IF XJ>0 THEN X0(I)=XJ MIN 199 ELSE X0(I)=1
2080 IF I=0 THEN GOTO 2220
2090 PRINT "~G'~C2";:PLOT 63+I,X0(I-1)+55,64+I,X0(I)+55
2100 GOTO 2220
2110 IF SC=0 THEN X1(I)=(XI*T/A2 MIN 199) MAX 1  :GOTO 2140
2120 XJ=MM*(1-(4/SC)*(1+(D2/80)-(XI/1600)))
2130 IF XJ>0 THEN X1(I)=XJ MIN 199 ELSE X1(I)=1
2140 IF I=0 THEN GOTO 2220
2150 PRINT "~G'~C4";:PLOT 63+I,X1(I-1)+55,64+I,X1(I)+55
2160 GOTO 2220
2170 IF SC=0 THEN X2(I)=(XI*T/A3 MIN 199) MAX 1  :GOTO 2200
2180 XJ=MM*(1-(4/SC)*(1+(D3/80)-(XI/1600)))
2190 IF XJ>0 THEN X2(I)=XJ MIN 199 ELSE X2(I)=1
2200 IF I=0 THEN GOTO 2220
2210 PRINT "~G'~C3";:PLOT 63+I,X2(I-1)+55,64+I,X2(I)+55
2220 NEXT
2230 NEXT
2240 IF LF=5 THEN PRINT CHR$(27);"OA2";CHR$(12); ELSE GOTO 2260
2250 PRINT CHR$(31);CHR$(31);"#";F1$;CHR$(27);"OA0";
2260 RETURN
2270 '
2280 'GRAPH LINES subroutine
2290 PRINT CHR$(27);"OA1";CHR$(12);
2300 PRINT "~G'~C7";:PLOT 64,55,64,255: PRINT "~C1";
2310 FOR N=0 TO 396 STEP 44
2320 PLOT 108+N,255,108+N,56
2330 NEXT
2340 ON SC+1 GOTO 2350,2350,2350,2430,2390
2350 FOR N=0 TO 180 STEP 20
2360 PLOT 65,75+N,511,75+N
2370 NEXT
2380 GOTO 2460
2390 FOR N=0 TO 175 STEP 25
2400 PLOT 65,80+N,511,80+N
2410 NEXT
2420 GOTO 2460
2430 FOR N=0 TO 167 STEP 33
2440 PLOT 65,88+N,511,88+N
2450 NEXT
2460 PRINT CHR$(27);"OA0";
2470 RETURN
2480 '
2490 FOR N=0 TO 4  'HOR AXIS CODE 0 INTERVAL subroutine
2500 IF N<4 THEN K=88*N ELSE K=348
```

6

```
2510 PRINT "[C7]U";:PLOT 140+K,54:
2520 PRINT USING "#.#";(N+1)/5
2530 NEXT
2540 VL=1
2550 RETURN
2560 '
2570 FOR N=0 TO 4  'HOR AXIS CODE 1 INTERVAL subroutine
2580 IF N<4 THEN K=88*N ELSE K=348
2590 PRINT "[C7]U";:PLOT 140+K,54:
2600 PRINT USING "#.#";.4+.4*N
2610 NEXT
2620 VL=2
2630 RETURN
2640 '
2650 FOR N=0 TO 4  'HOR AXIS CODE 2 INTERVAL subroutine
2660 PRINT "[C7]U";:PLOT 140+N*88,54:
2670 PRINT N+1
2680 NEXT
2690 VL=5
2700 RETURN
2710 '
2720 FOR N=0 TO 4  'HOR AXIS CODE 3 INTERVAL subroutine
2730 PRINT "[C7]U";:PLOT 140+88*N,54:
2740 PRINT USING "##";2*N+2
2750 NEXT
2760 VL=10
2770 RETURN
2780 '
2790 FOR N=0 TO 4  'HOR AXIS CODE 4 INTERVAL subroutine
2800 PRINT "[C7]U"; :PLOT 140+88*N,54:
2810 PRINT USING "##"; 4*N+4
2820 NEXT
2830 VL=20
2840 RETURN
2850 '
2860 FOR N=0 TO 4  'HOR AXIS CODE 5 INTERVAL subroutine
2870 PRINT "[C7]U";:PLOT 132+88*N,54:
2880 PRINT 10*N+10
2890 NEXT
2900 VL=50
2910 RETURN
2920 '
2930 ' START ADDRESS SEARCH THRU DATA IN BUFFER subroutine
2940 Q0=0
2950 FOR N=0 TO 500
2960 FOR K=0 TO 5
2970 Q=DA+100+K+6*N
2980 IF PEEK(Q)>127 THEN Q0=Q
2990 IF Q0>0 GOTO 3050
3000 NEXT
```

```
3010 NEXT
3020 IF Q0=0 THEN PRINT CHR$(12);"~C4";"Start Address ";
3030 PRINT "Not Found;  Buffer Data Missing or Defective."
3040 END
3050 RETURN
3060  '
3070 'PARAMETER CALC subroutine
3080 FOR N=0 TO 34
3090 NA=PEEK(Q0+2*N)
3100 S(N+1)=1-INT(64 AND NA)/64
3110 NEXT
3120 'DEL CALC
3130 DEL=0
3140 FOR N=0 TO 3
3150 SD(N)=8*S(4*N+1)+4*S(4*N+2)+2*S(4*N+3)+S(4*N+4)
3160 DEL=DEL+(10^(2-N))*SD(N)
3170 NEXT
3180 IF S(35)=1 THEN SC=3 ELSE SC=0
3190 IVL=4*S(17)+2*S(18)+S(19)
3200 F1=(4*S(20)+2*S(21)+S(22))*.25+.75
3210 F2=(4*S(23)+2*S(24)+S(25))*.5+1.5
3220 F3=4*S(26)+2*S(27)+S(28)+3
3230 B1=(((2*S(29)+S(30))*2) MAX 1)/4
3240 B2=(((2*S(31)+S(32))*2) MAX 1)/4
3250 B3=(((2*S(33)+S(34))*2) MAX 1)/4
3260 RETURN
3270  '
3280 'QUICK PLOT subroutine
3290 FOR I=0 TO 446 STEP ST
3300 FOR K=0 TO 2
3310 P=Q0+K*896+2*I+2
3320 NA=PEEK(P)
3330 NB=PEEK(P+1)
3340 CH=(NA AND 48)/16
3350 XI=(NA AND 15)*128+NB
3360 ON CH+1 GOTO 3370,3430,3490
3370 IF SC=0 THEN X0(I)=(XI*T/A1 MIN 199) MAX 1 :GOTO 3400
3380 XJ=MM*(1-(4/SC)*(1+(D1/80)-(XI/1600)))
3390 IF XJ>0 THEN X0(I)=XJ MIN 199 ELSE X0(I)=1
3400 IF I=0 GOTO 3540
3410 PRINT "~G'~C2";:PLOT 64+I-ST,X0(I-ST)+55,64+I,X0(I)+55:
3420 GOTO 3540
3430 IF SC=0 THEN X1(I)=(XI*T/A2 MIN 199) MAX 1  :GOTO 3460
3440 XJ=MM*(1-(4/SC)*(1+(D2/80)-(XI/1600)))
3450 IF XJ>0 THEN X1(I)=XJ MIN 199 ELSE X1(I)=1
3460 IF I=0 GOTO 3540
3470 PRINT "~G'~C4";:PLOT 64+I-ST,X1(I-ST)+55,64+I,X1(I)+55:
3480 GOTO 3540
3490 IF SC=0 THEN X2(I)=(XI*T/A3 MIN 199) MAX 1  :GOTO 3520
3500 XJ=MM*(1-(4/SC)*(1+(D3/40)-(XI/1600)))
```

8

```
3510 IF XJ>0 THEN X2(I)=XJ MIN 199 ELSE X2(I)=1
3520 IF I=0 GOTO 3540
3530 PRINT "~G'~C3";:PLOT 64+I-ST,X2(I-ST)+55,64+I,X2(I)+55:
3540 NEXT K
3550 NEXT I
3560 IF LF=5 THEN PRINT CHR$(27);"OA2";CHR$(12); ELSE GOTO 3580
3570 PRINT CHR$(31);CHR$(31);"*";F1$;CHR$(27);"OA0";
3580 RETURN
3590  '
3600 'RESCALING (linear) subroutine
3610 PRINT CHR$(27);"OA3";CHR$(12);"expansion factors ";
3620 PRINT "are:  1, 2, 4, 8, 16, 32.";
3630 PRINT CHR$(27);"OA0";
3640 PA=0
3650 PRINT CHR$(21);"~U144030";SPACE$(17)
3660 PRINT "~C2~1~U144030";"expand by";
3670 GOSUB 3940  '(scale exp. entry subroutine)
3680 IF Y<1 THEN PA=1
3690 IF Y>0 THEN A1=Y*A1 ELSE GOTO 3650
3700 PRINT "~U144030~2";"expand by ";Y$;"    "
3710 PRINT "~U272030";SPACE$(17);
3720 PRINT "~C4~1~U272030";"expand by";
3730 GOSUB 3940
3740 IF Y<1 THEN PA=1
3750 IF Y>0 THEN A2=Y*A2 ELSE GOTO 3710
3760 PRINT "~U272030~2";"expand by ";Y$;"    "
3770 MA= INT(1/A2) MIN INT(1/A1)
3780 PRINT "~U400030";SPACE$(14)
3790 PRINT "~C3~1~U400030";"expand by";
3800 GOSUB 3940
3810 IF Y<1 THEN PA=1
3820 IF Y>0 THEN A3=Y*A3 ELSE GOTO 3780
3830 MA=INT(1/A3) MIN MA
3840 A1=A1*MA :A2=A2*MA :A3=A3*MA
3850 PRINT "~U144030~2~C2";" A=  ";USING "#.#####          ";A1
3860 PRINT "~U272030~2~C4";" A=  ";USING "#.#####          ";A2
3870 PRINT "~U400030~2~C3";"  A=  ";USING "#.##### ";A3
3880 T=T*MA
3890 PRINT "~U000020~C7";"begins at"
3900 PRINT CHR$(27);"OA3";"~M~C0";CHR$(12);"~M~C6";
3910 PRINT CHR$(27);"OA0";"~U236042~C7";"microseconds";
3920 RETURN
3930  '
3940 'SCALE EXP. ENTRY subroutine
3950 INPUT Y$
3960 Y=0
3970 IF Y$="32" THEN Y=1/32
3980 IF Y$="16" THEN Y=1/16
3990 IF Y$="8" THEN Y=1/8
4000 IF Y$="4" THEN Y=1/4
```

```
4010 IF Y$="2" THEN Y=1/2
4020 IF Y$="1" THEN Y=1
4030 RETURN
4040  '
4050 'OVERALL SCALE FACTOR (linear) subroutine
4060 PRINT CHR$(27);"OA3";CHR$(12);
4070 PRINT "ENTER VERT. SCALE MULTIPLIER (0.5 to 2.0):";
4080 GOTO 4110
4090 PRINT CHR$(12);"ENTER VERT. SCALE MULTIPLIER ";
4100 PRINT "~N~1";"(0.5 to 2.0)";"~2";":";
4110 INPUT TD$
4120 FOR N=1 TO LEN(TD$) :H$=MID$(TD$,N,1)
4130 IF H$="." THEN 4150
4140 IF H$<"0" OR H$>"Z" OR H$>"9" AND H$<"A" THEN 4090
4150 NEXT
4160 TD=VAL(TD$)
4170 IF TD>2 OR TD<.5 THEN GOTO 4090
4180 PRINT "~M~C0";CHR$(12);"~M~C6";
4190 PRINT CHR$(27);"OA0";CHR$(21);"~U236042~C7";"microseconds"
4200 T=T*TD
4210 IF TD=1 THEN PB=0 ELSE PB=1
4220 RETURN
4230  '
4240 'ASSIGN FILE# subroutine
4250 PRINT CHR$(27);"OA3";CHR$(12);"ASSIGN FILE # ";
4260 W$=""
4270 INPUT "(8 characters):";F$
4280 FOR N=1 TO LEN(F$)
4290 H$=MID$(F$,N,1)
4300 IF H$<"0" OR H$>"Z" OR H$>"9" AND H$<"A" THEN 4320
4310 W$=W$+H$
4320 NEXT
4330 L=LEN(W$)-7
4340 IF L>0 THEN GOTO 4390
4350 FOR N=0 TO -L
4360 W$="0"+W$
4370 L=L+1
4380 NEXT
4390 F1$=MID$(W$,L,4)+"-"+RIGHT$(W$,4)
4400 F2$=RIGHT$(W$,8)
4410 PRINT CHR$(12);F1$;
4420 Z$=""
4430 INPUT "        ok for new file? - yes or no";Z$
4440 IF Z$="YES" THEN GOTO 4540
4450 IF Z$<>"NO" THEN GOTO 4410
4460 Z$="" :PRINT CHR$(12)
4470 PRINT "reassign #? - type ";"~N~C5";"FILE";"~N~C4";
4480 PRINT ", or if done? -press ";"~C5";"CR";"~C4";".";
4490 INPUT Z$
4500 IF Z$="FILE" GOTO 4250
```

10

```
4510 IF Z$<>"" GOTO 4460
4520 NF=0
4530 GOTO 4770
4540 PRINT CHR$(27);"OA2";CHR$(12);CHR$(31);CHR$(31);"#";F1$;
4550 GOSUB 4610
4560 POKE DA+20,1  'File Flag
4570 NF=1  '"new file" flag
4580 PRINT "~CO~K"; CHR$(28)
4590 RETURN
4600  '
4610  'SAY WHEN READY subroutine
4620 FOR K=1 TO 5
4630 PRINT CHR$(27);"OA3";CHR$(12);
4640 PRINT SPC(17);"~N~C4";"WHEN READY, PRESS ";"~N~C5";
4650 PRINT "CR";"~N~C4";".";
4660 FOR N=1 TO 100 :PRINT CHR$(7) :NEXT
4670 PRINT "~M~CO";CHR$(12);"~M~C6";
4680 PRINT "~?005";
4690 NEXT
4700 PRINT CHR$(12);SPC(17);"WHEN READY, PRESS ";"~N~C5";
4710 PRINT "CR";"~N~C4";".";
4720 PRINT "~?010"
4730 PRINT "~M~CO";CHR$(12);"~M~C6";
4740 PRINT CHR$(27);"OA0";CHR$(21);"~U236042~C7";"microseconds"
4750 PRINT CHR$(27);"OA1";
4760 INPUT Z$
4770 RETURN
4780  '
4790 'PRINT FILE # subroutine
4800 PRINT CHR$(27);"OA2";CHR$(12);CHR$(31);CHR$(31);"#";F1$;
4810 PRINT CHR$(27);"OA0";
4820 RETURN
4830  '
4840 PRINT  'MESSAGE PRIOR TO FULL PLOT subroutine
4850 PRINT CHR$(21);CHR$(12);
4860 PRINT "~C7";"The FULL PLOT will take about 3 minutes. ";
4870 PRINT "When it is complete, "
4880 PRINT "You may assign (or change) a FILE #, using 8 ";
4890 PRINT "alpha or numeric "
4900 PRINT "characters.  (If ";
4910 PRINT "you make an error here, note that only the "
4920 PRINT "LAST 8 characters are used.)
4930 PRINT
4940 PRINT "You may also request a new QUICK PLOT for scale ";
4950 PRINT "modification"
4960 PRINT "or you may terminate the processing ";
4970 PRINT "of the present data."
4980 PRINT "~?200";
4990 POKE DA+21,0  'Reset One-Time Flag
5000 RETURN
```

```
5010  '
5020 'RETRIEVE STORED PARAMETERS subroutine
5030 SC=PEEK(DA+27)
5040 IF SC>0 THEN GOTO 5130
5050 IF PEEK(DA+8)=0 THEN GOTO 5070
5060 A1=1/PEEK(DA+8)
5070 IF PEEK(DA+9)=0 THEN GOTO 5090
5080 A2=1/PEEK(DA+9)
5090 IF PEEK(DA+10)=0 THEN GOTO 5110
5100 A3=1/PEEK(DA+10)
5110 T=(256*PEEK(DA+11)+PEEK(DA+12))*1.25E-04
5120 GOTO 5140
5130 D1=PEEK(DA+28)-100 :D2=PEEK(DA+29)-100 :D3=PEEK(DA+30)-100
5140 F1=PEEK(DA+13)/4
5150 F2=PEEK(DA+14)/4
5160 F3=PEEK(DA+15)/4
5170 B1=PEEK(DA+16)/4
5180 B2=PEEK(DA+17)/4
5190 B3=PEEK(DA+18)/4
5200 DD=100*PEEK(DA+23)+10*PEEK(DA+24)+PEEK(DA+25)
5210 DEL=DD+.1*PEEK(DA+26)
5220 IVL=PEEK(DA+22)
5230 F2$=""
5240 FOR N=1 TO 8
5250 U=PEEK(DA-1+N)
5260 IF U<48 OR U>90 OR U>57 AND U<65 THEN GOTO 5310
5270 F2$=F2$+CHR$(PEEK(DA-1+N))
5280 NEXT
5290 F1$=LEFT$(F2$,4)+"-"+RIGHT$(F2$,4)
5300 RETURN
5310 F1$=" (none) "
5320 RETURN
5330  '
5340 'LOG RANGE CHANGE subroutine
5350 PRINT CHR$(27);"OA3";CHR$(12);
5360 PRINT "SELECT VERTICAL RANGE (1, 2, 3, or 4 DECADES):";
5370 INPUT LD$
5380 IF LD$="1" OR LD$="2" THEN GOTO 5400
5390 IF LD$="3" OR LD$="4" THEN GOTO 5400 ELSE GOTO 5350
5400 IF VAL(LD$)=SC THEN PB=0 ELSE PB=1
5410 SC=VAL(LD$)
5420 IF SC=3 THEN MM=198 ELSE MM=200
5430 PRINT "~M~C0";CHR$(12);"~M~C6";
5440 PRINT CHR$(27);"OA0";CHR$(21);"~U236042~C7";"microseconds"
5450 RETURN
5460  '
5470 'LOG WAVEFORM SHIFT subroutine
5480 PRINT CHR$(27);"OA3";CHR$(12);
5490 PRINT"  SELECT + or - INTEGER db VALUES FOR UP/DOWN SHIFTS"
5500 PRINT CHR$(27);"OA0";
```

12

```
5510 PA=0
5520 PRINT CHR$(21);"~U144030";SPACE$(17)
5530 PRINT "~C2~1~U144030";"shift by";
5540 INPUT XY$
5550 IF LEN(XY$)>3 THEN GOTO 5520 ELSE XY=VAL(XY$)
5560 IF XY>80 OR XY<-80 THEN GOSUB 5830 :GOTO 5520 '(err. msg.)
5570 D1=D1-XY
5580 IF XY<>0 THEN PA=1
5590 PRINT "~U144030~2";"shift by ";XY;"     "
5600 PRINT "~U272030";SPACE$(17);
5610 PRINT "~C4~1~U272030";"shift by";
5620 INPUT XY$
5630 IF LEN(XY$)>3 THEN GOTO 5600 ELSE XY=VAL(XY$)
5640 IF XY>80 OR XY<-80 THEN GOSUB 5830 :GOTO 5600 '(err. msg.)
5650 D2=D2-XY
5660 IF XY<>0 THEN PA=1
5670 PRINT "~U272030~2";"shift by ";XY;"     "
5680 PRINT "~U400030";SPACE$(14)
5690 PRINT "~C3~1~U400030";"shift by";
5700 INPUT XY$
5710 IF LEN(XY$)>3 THEN GOTO 5680 ELSE XY=VAL(XY$)
5720 IF XY>80 OR XY<-80 THEN GOSUB 5830 :GOTO 5680 '(err. msg.)
5730 D3=D3-XY
5740 IF XY<>0 THEN PA=1
5750 PRINT "~U144030~2~C2";" D=  ";USING "+## db        ";D1
5760 PRINT "~U280030~2~C4";"D=  ";USING "+## db        ";D2
5770 PRINT "~U416030~2~C3";"D=  ";USING "+## db  ";D3
5780 PRINT "~U000020~C7";"begins at"
5790 PRINT CHR$(27);"OA3";"~M~C0";CHR$(12);"~M~C6";
5800 PRINT CHR$(27);"OA0";"~U236042~C7";"microseconds";
5810 RETURN
5820  '
5830 'EXCESS SHIFT MSG subroutine
5840 PRINT CHR$(27);"OA3";CHR$(12);"~N~1"
5850 PRINT "         SHIFT MAGNITUDE IS LIMITED TO 80 db.";
5860 PRINT "~?050";CHR$(12);"~2"
5870 PRINT"  SELECT + or - INTEGER db VALUES FOR UP/DOWN SHIFTS"
5880 PRINT CHR$(27);"OA0";
5890 RETURN
```

```
7990 '**************************************************
8000 'This sequence for key F6. (STEP 8000; Do not renumber.)
8010 PRINT CHR$(12)
8020 POKE (256*PEEK(15175)+PEEK(15174)+19),6
8030 RUN
8040  '
8990 '**************************************************
9000 'This sequence for key F4. (STEP 9000; Do not renumber.)
9010 PRINT CHR$(12)
9020 PRINT CHR$(27);"ROF"        31250 Baud rate
9030 DEFUSR0=17000
9040 POKE 17100,255
9050 AA=USR0(ZZ)
9060 IF PEEK(17100)=255 THEN GOTO 9120
9070 PRINT "~N~C4";"FILTER-DIGITIZER IS NOT RESPONDING ";
9080 PRINT "PROPERLY."  :PRINT
9090 PRINT "Correct problem and press key ";"~C5";"F4";
9100 PRINT "~C4";" again."
9110 END
9120 POKE (256*PEEK(15175)+PEEK(15174)+19),4     put 4 into 32768+19
9130 RUN
9140  '
9150 'END OF PROGRAM GRAPH
```

SECTION C-6. GRANAFEX

```
100 'THIS IS PROGRAM GRAFANEX
110  '
120 GOSUB 190  '(store away parameters)
130 GOSUB 540  '(print out param)
140 IF NF=1 THEN GOSUB 850  '(file on disk)
150 GOSUB 1450  '(final message)
160 DOS"LOAD GRAPH/1"
170 END
180  '
190 'STORE AWAY PARAMETERS subroutine
200 POKE DA+8,1/A1
210 POKE DA+9,1/A2
220 POKE DA+10,1/A3
230 TT=1000*T/32
240 POKE DA+11,INT(TT)
250 PT=8000*T   :QT=INT(PT+.5) MOD 256
260 POKE DA+12,QT
270 POKE DA+13,4*F1
280 POKE DA+14,4*F2
290 POKE DA+15,4*F3
300 POKE DA+16,4*B1
310 POKE DA+17,4*B2
320 POKE DA+18,4*B3
330 POKE DA+19,6
340 POKE DA+22,IVL
350 L1=INT(DEL/100)
360 L2=INT((DEL-100*L1)/10)
370 L3=INT(DEL-100*L1-10*L2)
380 L4=INT((DEL-100*L1-10*L2-L3)*10)
390 POKE DA+23,L1
400 POKE DA+24,L2
410 POKE DA+25,L3
420 POKE DA+26,L4
430 POKE DA+27,SC
440 POKE DA+28,100+D1
450 POKE DA+29,100+D2
460 POKE DA+30,100+D3
470 IF F2$="" THEN F2$=" (none) "
480 FOR N=1 TO 8
490 Q$=MID$(F2$,N,1)
500 POKE DA-1+N,ASC(Q$)
510 NEXT
520 RETURN
530  '
540 'PRINT OUT PARAMETERS subroutine
550 PRINT CHR$(27);"OA0";CHR$(21);CHR$(12);CHR$(28);"~C7"
560 PRINT "FILE # ";
570 FOR N=1 TO 8
580 IF N=5 AND F2$<>" (none) " THEN PRINT "-";
590 PRINT CHR$(PEEK(DA-1+N));
600 NEXT
610 SC=PEEK(DA+27)
620 IF SC=0 THEN GOTO 680
630 PRINT "         Logarithmic range =";SC;"decades"
640 D1=PEEK(DA+28)-100   :PRINT "D1=";D1;
650 D2=PEEK(DA+29)-100   :PRINT "   D2=";D2;
```

PROGRAM GRAFANEX CONTINUED

```
660 D3=PEEK(DA+29)-100  :PRINT "   D3=";D3
670 GOTO 730
680 FA=(256*PEEK(DA+11)+PEEK(DA+12))/1000
690 PRINT "      Linear Scale factor = ";FA
700 PRINT "A1=";1/PEEK(DA+8);
710 PRINT "   A2=";1/PEEK(DA+9);
720 PRINT "   A3=";1/PEEK(DA+10)
730 PRINT "F1=";PEEK(DA+13)/4;
740 PRINT "   F2=";PEEK(DA+14)/4;
750 DD=100*PEEK(DA+23)+10*PEEK(DA+24)+PEEK(DA+25)
760 DEL=DD+.1*PEEK(DA+26)
770 PRINT "   F3=";PEEK(DA+15)/4,"Delay=";DEL
780 PRINT "B1=";PEEK(DA+16)/4;
790 PRINT "   B2=";PEEK(DA+17)/4;
800 IVL=PEEK(DA+22)
810 IF IVL<3 THEN VL=IVL^2+1 ELSE VL=10*((IVL-3)^2+1)
820 PRINT "   B3=";PEEK(DA+18)/4,"Interval=";VL
830 RETURN
840 '
850 'FILE ON DISK subroutine
860 GOSUB 1350  '(calc avail disk space subroutine)
870 IF AF=-1 THEN PRINT ELSE GOTO 900
880 PRINT "~C4";"Check Drive 2 for problem!"
890 GOTO 930
900 PRINT "Disk Drive #2 available file spaces:";AF;"."
910 IF AF>0 GOTO 1000 ELSE PRINT
920 PRINT "~C4";"Change disk in Drive 2; no space remains."
930 PRINT "To resume, type ";"~C5";"GO";"~C4";" when ready.";
940 PRINT "  To cancel filing, press ";"~C5";"CR";"~C4";"."
950 INPUT Z$
960 IF Z$<>"GO" THEN NF=0  :GOTO 1330
970 GOSUB 540  '(print out param)
980 GOTO 850
990 GOTO 1330
1000 PRINT "~C6" :PRINT "WAIT; filing in process."
1010 ON ERROR GOTO 1070
1020 DOS"STORE "+F2$+"/2 "+HEX$(DA)+" "+HEX$(DA+5475)+"
1030 DOS"COMPRESS/2"
1040 DOS"FETCH "+F2$+"/2 "+HEX$(DA)+"
1050 EF=0  :NF=1
1060 GOTO 1330
1070 'Disk error trap sequence
1080 NF=0  :E$=""  :PRINT "~R"
1090 IF ERR=70 GOTO 1150
1100 IF ERR=71 GOTO 1160
1110 IF ERR=82 GOTO 1170
1120 IF ERR=83 GOTO 1180
1130 IF ERR=119 GOTO 1190
1140 GOTO 1200
1150 E$="#14."  :GOTO 1220
1160 E$="#15."  :GOTO 1220
1170 E$="#20."  :GOTO 1220
1180 E$="#21."  :GOTO 1220
1190 E$="#45."  :GOTO 1220
1200 E$="AS NOTED BELOW."
```

PROGRAM GRAFANEX CONTINUED

```
1210 EF=1
1220 PRINT "~C4" :PRINT "DISK SYSTEM HAS REJECTED REQUEST ";
1230 PRINT "TO FILE #";F1$;","
1240 PRINT "CLAIMING DOS ERROR ";E$  :PRINT
1250 IF EF=1 GOTO 1060
1260 PRINT "~C6";"Correct problem and type ";"~C5";"GO";
1270 PRINT"~C6";".  To omit filing, press ";"~C5";"CR";"~C6";"."
1280 K$=""
1290 INPUT K$
1300 IF K$="" THEN GOTO 1330
1310 IF K$="GO" THEN RESUME 850
1320 PRINT  :GOTO 1260
1330 RETURN
1340  '
1350 'CALC OF AVAIL DISK SPACE subroutine
1360 ON ERROR GOTO 1410
1370 DOS"LDDIR /2 V$"
1380 AS=VAL("&H"+RIGHT$(V$(0),4))
1390 AF=INT(AS/43)
1400 GOTO 1430
1410 AF=-1
1420 RESUME 1430
1430 RETURN
1440  '
1450 PRINT ' FINAL MESSAGE subroutine
1460 PRINT
1470 PRINT "~C6";"PROCESSING OF THE DATA HAS BEEN COMPLETED."
1480 PRINT
1490 PRINT "The parameters listed above have been stored in ";
1500 PRINT "buffer memory";
1510 IF NF=1 THEN GOTO 1610
1520 IF PEEK(DA+19)=5 THEN GOTO 1580
1530 PRINT "," :PRINT "temporarily. To recall the data for ";
1540 PRINT "processing, press key ";"~C5";"F6";"~C6";"."
1550 PRINT
1560 PRINT "Otherwise, a new SNAPSHOT or FILE may be ";
1570 GOTO 1650
1580 PRINT :PRINT "temporarily, while the entire data set ";
1590 PRINT "remains in disk file."
1600 GOTO 1640
1610 PRINT :PRINT "temporarily, and the entire data set ";
1620 PRINT "has been permanently " :PRINT "filed on DisK."
1630 PRINT
1640 PRINT :PRINT "Now, a new SNAPSHOT or FILE may be ";
1650 PRINT "called using keys ";"~C5"
1660 PRINT "F4";"~C6";", or ";"~C5";"F5";"~C6";", respectively."
1670 PRINT  :PRINT
1680 IF PEEK(DA+20)=0 AND NF=1 THEN GOTO 1700
1690 GOTO 1720
1700 PRINT "~C4";"FILE FLAG ERROR:  flag=";PEEK(DA+20);
1710 PRINT "     NF=";NF
1720 IF EF=1 THEN GOSUB 1000 '(last part of file on disk subr.)
1730 RETURN
1740 'END OF PROGRAM GRAFANEX
```

FIL

7 -